Design method development and software implementation for fibre-reinforced concrete slabs-on-ground

by

Frederik Jacobus Mudge

Thesis presented in fulfilment of the requirements for the degree of Master of Engineering in Civil Engineering in the Faculty of Engineering at Stellenbosch University

Division of Structural Engineering,
Department of Civil Engineering,
University of Stellenbosch,
Private Bag X1, Matieland 7602, South Africa.

Supervised by:
Prof. WP Boshoff
Dr. GC Van Rooyen

December 2017

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Signature: ....

Date: December 2017

Abstract

Ground-supported concrete slabs are common structural elements, used for a multitude of purposes. In industrial flooring applications, slabs-on-ground (SOG) are often subjected to severe loads, concentrated at points or acting over extended areas. Adequate reinforcement of such slabs is essential to obtain sufficient load capacity and to guarantee serviceability of a slab throughout its lifetime.

Synthetic fibre reinforcement has been shown to be effective in increasing the tensile strength and toughness of concrete slabs-on-ground. It increases the load capacity of slabs without requiring procurement of costly steel-mesh, the labour associated with installing it, or major alterations to concrete mix design.

Although extensive research has been carried out to analyse and predict the performance of synthetic-fibre reinforced concrete (SynFRC) slabs-on-ground, no universally accepted design guideline exists. Similarly, no computer-based design packages that facilitate the analysis and design of such slabs are available.

In this study a comprehensive set of algorithms is developed for the analysis and design of SynFRC ground-supported slabs. It includes an algorithm that can optimise any given slab design in terms of cost. The proposed algorithms are based on an extensive review of relevant academic and industrial literature pertaining to SynFRC, slabs-on-ground and their associated design approaches. Long term settlement and the bearing capacity of soil are not accounted for. The reaction of soil to slab loading is included by means of a modulus of subgrade reaction, k. The yield-line approach to assessing point load capacities is adopted, while elastic methods are employed to analyse the effect of line- and uniformly distributed loads on the structure.

A software prototype that implements the algorithms and provides a user friendly interface is developed using the Java programming language. It includes various features which aid the process of modelling a slab, such as the generation of the most adverse wheel loads within a traffic zone.

To ensure the validity of all algorithms and their implementation, a series of unit tests and validations are carried out.

It is concluded that the proposed algorithms and software prototype operate successfully and yield useful results.

Opsomming

Grondgesteunde betonblaaie is algemene struktuurelemente wat vir 'n verskeidenheid doeleindes gebruik word. In industriële vloertoepassings word grondgesteunde betonblaaie dikwels aan relatief hoë belastings blootgestel. Die belastings kan in die vorm van gekonsentreerde puntlaste of verspreide laste oor 'n area wees. Geskikte versterking van sulke blaaie is noodsaaklik om voldoende laaikapasiteit te verkry en die diensbaarheid van 'n blad te verseker oor die leeftyd daarvan.

Dit is al bewys dat sintetiese veselversterking effektief is om die treksterkte en taaiheid van betonblaaie op die grond te verhoog. Dit verhoog die laaikapasiteit van blaaie en skakel die voorsiening van staal maas versterking, teen 'n relatief hoë koste met die arbeid geassosieer met die installering daarvan, of groot veranderinge aan betonmengselontwerp uit.

Alhoewel uitgebreide navorsing al gedoen is om die gedrag van sintetiese veselversterkte betonblaaie op die grond te analiseer en te voorspel, bestaan daar geen universeel aanvaarde ontwerp riglyn nie. Daar is ook geen rekenaargebaseerde ontwerppakkette wat die ontleding en ontwerp van sulke blaaie ondersteun nie.

In hierdie studie word 'n omvattende stel algoritmes ontwikkel vir die analise en ontwerp van sintetiese veselversterkte grondgesteunde betonblaaie. Dit sluit 'n algoritme in wat 'n gegewe bladontwerp in terme van koste kan optimeer. Die voorgestelde algoritmes is gebaseer op 'n uitgebreide oorsig van relevante akademiese en industriële literatuur met betrekking tot veselversterkte beton, betonblaaie op die grond en hul gepaardgaande ontwerpbenaderings. Langtermyn versakking en die evalueer van die dravermoë van die grond ondersteuning word nie in hierdie studie in ag geneem nie. Die reaksie van grond tot bladbelasting word ingesluit deur middel van 'n stutlaag reaksiemodulus, k. Die sogenaamde “yield-line” benadering vir die assessering van puntladingkapasiteite word aangeneem, terwyl elastiese metodes gebruik word om lyn- en uniform verspreide laste te analiseer.

'n Prototipe rekenaar toepassingsprogram, wat die algoritmes implementeer en 'n gebruikersvriendelike koppelvlak voorsien, word ontwikkel met behulp van die Java programmeringstaal. Dit bevat verskeie eienskappe wat die modellering van 'n blad vergemaklik, soos die opwekking van die mees ongunstige wielbelastings in 'n verkeersone.

Om die geldigheid van al die algoritmes en hul implementering te verseker, word 'n reeks eenheidstoetse en validasies uitgevoer.

Daar word tot die gevolgtrekking gekom dat die voorgestelde algoritmes en prototipe rekenaar toepassingsprogram suksesvol werk en nuttige resultate lewer.

Acknowledgements

As the author of this thesis, I would like to express my sincere gratitude towards the following people, who in some way contributed to the completion of this project.

- My family, and in particular my parents, Jaco and Estelle Mudge, for their care, love and support over the past 24 years.
- Prof. Billy Boshoff, for giving me the opportunity to work on this project, and for his essential guidance in researching the necessary theoretical concepts.
- Dr. Gert Van Rooyen, for his continued valuable support in research, algorithm and software development and report writing.
- Ms. Nelmari Harmzen, for her unconditional love and support over the duration of this project.
- My friends and classmates, for their companionship and support over the past two years.
- Lithon Project Consultants, for supporting my ambitions through partial funding of my education, over the past four years.

Contents

List of figures

List of tables

Nomenclature

A	Surface area of a specific bay [m^2]
a	Equivalent load radius [mm]
A_c	Shear face area, at the critical perimeter [mm^2]
A_f	Shear face area, at the load edge [mm^2]
b	Width of beam specimen [mm]
C	Current load capacity of a specific bay, considering the governing load type
C_c	Unit cost of concrete, per cubic meter
C_f	Unit cost of fibre reinforcement, per kilogram
C_T	Total cost of a slab
d	Effective depth of slab/bay cross section [mm]
E	Modulus of elasticity of concrete [N/mm^2]
f_{ck}	Characteristic cylinder compressive strength of concrete at 28 days [MPa]
$f_{ctd,fl}$	Design flexural tensile strength of concrete [MPa]
f_{ctm}	Mean axial tensile strength of concrete [MPa]
fD	Fibre dosage of concrete [kg/m^3]
F_{R}	Applied flexural load at a specific CMOD [N]
f_{R1}	Residual flexural tensile strength corresponding to a CMOD of 0.5 mm [MPa]
f_{R4}	Residual flexural tensile strength corresponding to a CMOD of 3.5 mm [MPa]
h	Slab/bay thickness [mm]
h_c	Crack height of a fibre-reinforced concrete section in flexure [mm]
h_{sp}	Depth of a beam section, from the top of the specimen to the notch tip [mm]
h_{ux}	Neutral axis depth of a fibre-reinforced concrete section in flexure [mm]
k	Modulus of subgrade reaction [N/mm^2/mm]
l	Radius of relative stiffness [mm]
M	Magnitude of the governing load on a specific bay
M_u	Moment capacity of a fibre-reinforce concrete section [kNm]
M_{un}	Moment capacity of a plain concrete section [kNm]

N	Compressive force [N]
P_c	Perimeter length of a combined point load, at the critical perimeter [mm]
P_{f}	Perimeter length of a combined point load, at the face of the loaded area [mm]
P_{lin}	Ultimate capacity under line loading [kN/m]
$P_{p,max}$	Maximum load capacity in punching, considering the face of the loaded area [N]
P_{p}	Maximum load capacity in punching, considering the critical perimeter [N]
P_{u}	Ultimate capacity under concentrated loading [kN]
q	Ultimate capacity under uniform distributed loading [kN/m^2]
r	Modified load radius [mm]
s	Span of beam specimen [mm]
T	Tensile force [N]
u_{f}	Perimeter length of a single point load, at the face of the loaded area [mm]
u_{c}	Perimeter length of a single point load, at the critical perimeter [mm]
v	Poisson's ratio for concrete, taken as 0.2
v_{fib}	Increase in shear strength provided by reinforcement [MPa]
v_{max}	Maximum allowable shear stress [MPa]
$v_{Rd,c}$	Minimum shear strength of unreinforced concrete [MPa]

List of Greek symbols

γ_m	Partial safety factor for material properties
γ_F	Partial safety factor for loads
λ	Slab/bay characteristic value [mm^{-1}]
$\sigma_{\mathrm{r}1}$	Mean axial tensile strength corresponding to a CMOD of 0.5 mm [MPa]
$\sigma_{\mathrm{r}4}$	Mean axial tensile strength corresponding to a CMOD of 3.5 mm [MPa]

List of acronyms

ACI	American Concrete Institute
CMOD	Crack mouth opening displacement
FRC	Fibre reinforced concrete
GUI	Graphical user interface
HyFRC	Hybrid fibre reinforced concrete
MHE	Material handling equipment
MOR	Modulus of rupture
MSFRC	Macro-synthetic fibre reinforced concrete
SFRC	Steel fibre reinforced concrete
SOG	Slab-on-ground
SynFRC	Synthetic fibre reinforced concrete
TR34	Technical Report 34 (Concrete Society 2013)
TR65	Technical Report 65 (Concrete Society 2007)
TZ	Traffic Zone
UDL	Uniform Distributed Load
ULS	Ultimate Limit State
UML	Unified Modelling Language
VNA	Very Narrow Isle

Glossary

Bay	A continuous segment of a slab, typically cast without interruption.
Bay-region	An area on a specific bay which has a unique and independent load capacity when considering a certain load type.
Combined point load	Two or more point loads which are spaced sufficiently close together to be considered as a single point load.
Dual point load	Two single point loads which are spaced sufficiently close together to have a combined effect on a slab, but which are too far apart to be considered as a combined point load.
Line load	A load acting on an area narrow enough to be approximated as a straight line.
Line load segment	A segment of a line load which acts within or across a single bay region
Point load	A load acting on a single, relatively small, concentrated area.
Quadruple point load	Four single point loads which are spaced sufficiently close together, in a rectangular configuration, to have a combined effect of a slab, but which are too far apart to be considered as a combined point load.
Single point load	A point load which has an effect that is independent of other point loads.
Slab	A large, flat concrete structure, consisting of multiple bays separated by joints.
Traffic zone	A region on the slab over which a particular vehicle is likely to move.

Programming terminology

Abstract method	A method, specified by a specific super-class, which is to be completed by all of its sub-classes.
Attribute	A characteristic value/object of a specific type, assigned to an object.
Class	A template for creating virtual objects, specifying all relevant required attributes and functionalities.
Inheritance	The process by which attributes or functionalities of a super-class are automatically applied to all of its sub-classes.
Instance	An example or single manifestation of something.
Instantiation	The process of creating an instance.
Method	A segment of code which performs a specific operation. Each method is assigned a specific visibility, name and expected datatype to be returned. Input values and types required by each method are also pre-defined.
Object	An instance of a specific class, with unique and independent characteristics conforming to the template set out by the class.
Sub-class	An extension of a certain super-class. A sub-class can have unique attributes and functionalities and objects of such a class can be instantiated.
Super-class	A class which represents an abstract concept and has one or more sub-classes, which conform to certain criteria set out by the super-class. Objects of super-classes cannot be instantiated, but objects of their sub-classes can.

Chapter 1

Introduction

1.1. Background information

In the fields of civil engineering and construction, slabs-on-ground (SOG), also referred to as ground-supported slabs or slabs-on-grade, are a common structural element. Even though these slabs appear to be relatively simple in terms of their behaviour and design, the variable nature of soil, combined with the wide variety of possible slab purposes and specifications complicate their design.

Over the past several decades, various methods of reinforcement have been proposed and investigated to increase the performance of slabs. These methods include steel fabric mesh, post-tensioned rods and various types of fibres. Each of these methods have been shown to have distinct advantages and disadvantages, depending on the specific situation under consideration. For slabs-on-ground, fibre reinforcement is particularly attractive, as it allows relatively easy and cost-effective construction, while providing ample moment and shear resistance.

No formal design codes, which outline appropriate methods of analysing and designing fibre-reinforced slabs-on-ground, are currently available. Therefore, simplified and conservative estimates and rules of thumb are commonly relied on for manual slab-on-ground thickness design and reinforcement specification. For the same reason, the vast majority of structural design software packages do not support the use of fibres as reinforcing in ground-supported slabs.

1.2. Objectives of the study

The first main objective of this project, denoted Objective A, is to develop a set of slab-design algorithms which can be used for the analysis and design of ground supported slabs reinforced with synthetic fibres - more specifically, CHRYSO®Fibre S50 fibres. Design outcomes must subscribe to industrial loading conditions and performance requirements.

In order to achieve this objective, the following secondary objectives are completed during this study:

A1- Data collection

Collection and compilation of relevant academic and industrial findings, regarding slabs-on-ground, fibre reinforced concrete and other appropriate concepts.

A2- Soil investigation

Gaining a reasonable understanding of the behaviour of soils as supporting structures to slabs-on-ground.

A3- Design method examination

Examination and comparison of popular design methods and their fundamental concepts.

A4- Algorithm development

Development of versatile and comprehensive slab analysis and design algorithms, based on the analysis procedures outlined by Technical Report 34 (Concrete Society 2013).

The second main objective, denoted Objective B, is to deliver a software prototype based on an object-oriented slab model, which implements the Analysis and Design algorithms compiled during completion of Objective A.

B1- Analysis and design model development

Development of effective object models, which represent the various components of slabs-on-ground and their associated loads. For analysis purposes, allow for the computation of all relevant load interaction and capacity values. For design purposes, incorporate methods of approximating combinations of slab characteristic values which will provide sufficient load capacity.

B2- Prototype software creation

Software implementation of the slab-design algorithms, creating a simple, user friendly interface.

Finally, the third main objective of the project, denoted Objective C, is to add an optimisation functionality to the software prototype.

C- Optimisation

Development and implementation of an optimisation feature, which will enable the software to deliver the most cost-effective possible solution to a certain design situation, in terms of materials used.

1.3. Scope

The following concepts could ultimately influence the design of fibre-reinforced slabs-on-ground. However, they are not considered directly relevant and are therefore omitted from the scope of this study:

- Slabs-on-ground which are carried by piles
- Design of dowels and other load-transfer devices
- Design of slabs that are resistant to high heat resulting from exposure to fire
- Possible interaction of slabs-on-ground with destructive chemicals
- The presence and intensity of wind during slab construction
- Compaction and consolidation of the soil underneath the slab
- Steel reinforcing systems in slabs
- Required level of slab cleanliness
- Concrete mix designs
- Post-tensioned slab reinforcing systems

1.4. Methodology

A systematic approach, corresponding with sub-objectives set out in Section 1.2, to achieve a versatile slab-design algorithm and software is as follows:

A1- Compilation of a full literature review, gathering and appropriately organising all relevant findings from past research, experiments and the structural engineering and construction industry.

A2- Investigation of literature regarding the soil and support structures underneath slabs-on-ground, highlighting important factors to consider during design.

A3- Examination of the basic and conservative elastic design method, the more advanced yield-line method, as well as their applications to the design of synthetic fibre reinforced concrete (SynFRC) slabs-on-ground.

A4- Considering and adapting the steps outlined in TR34, develop a complete Design Algorithm in various stages, as described in Chapter 3.

Similarly, the methodology for achieving Objectives B and C is as follows:

B1- By decomposing a generalised slab-on-ground setup into its basic components, identify and set-up programmable objects to represent the slab and its associated loads. Incorporate functionalities which allow the computation and/or approximation of all necessary capacity and design data into the various object models.

B2- Using the Java programming language, develop prototype software which gathers input data from a user, performs appropriate calculations and delivers a suitable solution.

C- Examine existing methods of software process optimisation and identify appropriate options. Apply the most suitable technique and develop a suitable Optimisation Algorithm. Appropriately adapt all design algorithms and software implementation.

Chapter 2

Literature review

2.1 Introduction to literature review

In recent years, numerous research papers and design guidelines have been published which, on the basis of academic investigation and experimental/industrial experience, describe possible ways to approach the analysis and design of slabs-on-ground. Possibly the most highly regarded of these guides was compiled by The Concrete Society (2013) and is titled “Technical Report 34, Concrete industrial ground floors, a guide to design and construction”, commonly referred to as TR34. Other examples of such papers include “Load carrying capacity of concrete pavements” (Meyerhof 1962), “Yield-line analysis of slabs-on-grade” (Baumann, Weisgerber 1983) and “Practical yield line design” (Kennedy, Goodchild 2004).

This chapter serves to compile useful information from the above sources, supplemented by an array of other academic sources, in order to gain a comprehensive understanding of the behaviour of fibre reinforced concrete (FRC) slabs-on-ground. This will aid the compilation of a versatile slab-design algorithm, which can be implemented by means of computerised methods. Due consideration is also given to past research, experiments and the structural engineering and construction industry.

In order to gain a comprehensive understanding of FRC slabs-on-ground, the topic is dissected and its components individually examined throughout the respective sub-sections of this chapter, keeping the scope of the project in mind.

2.2 Concrete slabs-on-ground

2.2.1 Requirements for concrete industrial ground slabs

The requirements of any individual slab are largely dependent on its specific characteristics, situation and purpose. However, many general requirements exist which are widely applicable in the field of concrete slabs-on-ground.

The requirements for concrete industrial ground floors, as outlined by TR34 (Concrete Society 2013), are as follows:

- The ability to carry all types of loads, without unacceptable cracking, deflection, settlement or joint damage
- Serviceability of the floor throughout its lifetime, under reasonable maintenance and loading conditions
- Sufficient design and optimal layout of joints
- Suitable surface regularity
- Proper resistance to abrasion, slip and chemical attack
- An appropriate finish

Surface regularity is considered especially relevant. The preservation of surface regularity is essential in floors which are to be utilised along with high-lift materials handling equipment (MHE), as it will directly influence the precision with which the MHE can operate. Two aspects of regularity are flatness, i.e. the absence of height variations over short distances, and levelness, i.e. the absence of height variations over longer distances. These two concepts are illustrated in Figure 2.1.

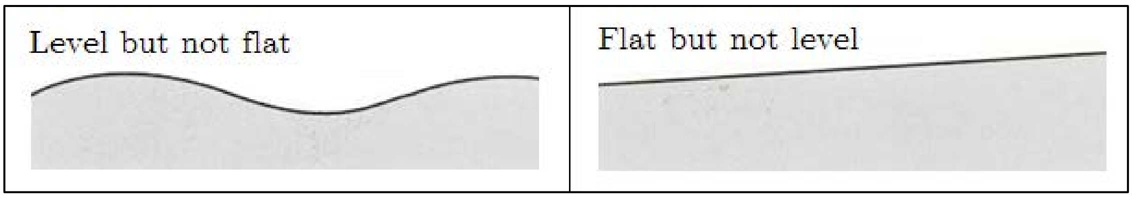


Figure 2.1: Illustration of flatness and levelness (Concrete Society 2013)

A further set of requirements can be deduced from prevention of the following common and undesirable slab behaviour traits (Concrete Society 2013):

Crazing: An irregular pattern of fine surface cracks, which have negligible structural effects but an undesirable visual impact.

Curling: The tendency of a slab to curve in a convex upward manner, due to differential shrinkage of the respective concrete layers. This can have various adverse effects, including cracking and loss of sub-base support.

Delamination: The detachment of a thin layer of material from the slab surface, caused by bleed water beneath the surface and other less common factors, such as differential setting of the surface, air entrainment, bleed characteristics of the concrete and the application of dry-shake toppings.

Surface protrusion of aggregate or fibres is typically also considered objectionable. However, TR65 (Concrete Society 2007) argues that protruding synthetic fibres quickly wear away and will not cause damage to vehicle tyres or pedestrians, and are therefore not considered to be catastrophic.

Various other factors, such as extreme heat/cold, wind or rain could demand that special consideration be given to specific slab requirements. This may result in a need for a more controlled construction environment. Thus, construction sequence considerations are important to determine whether transient environmental factors could influence design (ACI 2010).

From a design perspective, the most important requirement to consider is that all loads must be carried without reducing the serviceability of a slab. Seeing as all other requirements rely on correct construction and maintenance, they are not considered further.

2.2.2 Loading of concrete industrial ground slabs

The loads acting on a slab are usually the governing design consideration. Accordingly, sufficient care should be taken to ensure all loads are accounted for, as accurately as possible. This does not only involve load intensity, size and location, but also factors such as the expected number of repetitions of a certain load, load duration and interaction between loads and edges (ACI 2010).

Typical loads

Typical loads on industrially used concrete slabs include (Concrete Society 2013):

- Static loads
  - o Pallet racking equipment (see Figure 2.2)
  - o Material stacking directly on the slab

Figure 2.2: Typical back-to-back pallet racking configuration (Concrete Society 2013)

- Dynamic loads
  - o Warehouse equipment/MHE (see Figure 2.3)
    - a. Pallet trucks
    - b. Counterbalance trucks
    - c. Reach trucks
    - d. Front and lateral stackers and VNA trucks
    - e. Articulated counterbalance trucks
    - f. Stacker cranes

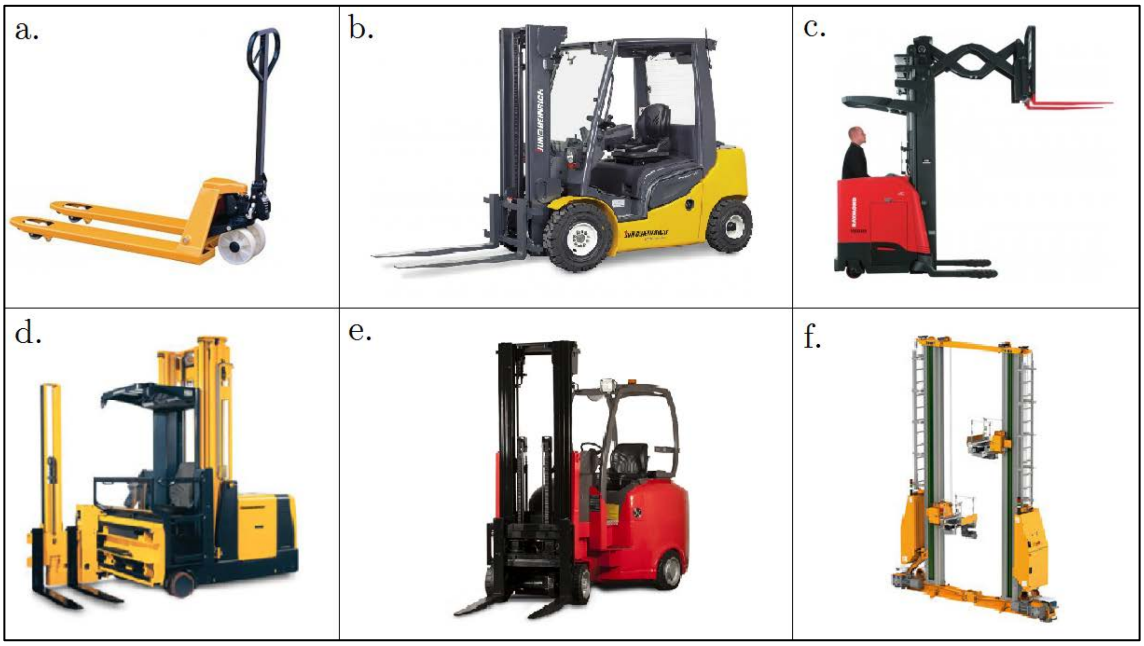


Figure 2.3: Various types of warehouse equipment/MHE
a. Pallet truck (Pallettruck Warehouse 2017)
b. Counterbalance truck (Bridge-end training facility 2017)
c. Reach truck (Raymond handling concepts 2017)
d. Front and lateral stacker (Industry plaza 2017)
e. Articulated counterbalance truck (Mentor training 2017)
f. Stacker crane (Indiamart 2017)

Load distribution

All loads acting on a slab can be categorised as either a point, line or uniform distributed (UDL) load. These ideal terms are commonly used in the fields of engineering and science. However, since point and line loads, by definition, have dimensions which are infinitesimally small and therefore unrealistic, some description of the terms for the purpose of this study is warranted.

In order to calculate the stresses imposed by a point load, a simplification is made to effectively quantify its actual dimensions. The most common point loads acting on industrial concrete slabs-on-ground are column baseplates and MHE wheels. While wheel loads are generally approximated as being circular, most column baseplates are rectangular. An equivalent load radius, a, can be easily calculated which would produce a circle of the same surface area. For example, considering a load with width, w, and breadth, b, the equivalent radius would be calculated using Equation 1.

$$a = \sqrt{\frac{w \cdot b}{\pi}} \qquad \text{Eq. (1)}$$

If, when considering pneumatic wheel loads, contact area details are unknown, the surface area can simply be calculated as: $\mathrm{A} = \frac{\text{wheel load}}{\text{tyre pressure}}$. This method, however, is considered to be conservative as the effect of tyre-wall tension is neglected (ACI 2010). For other wheel types, the manufacturer should be consulted for load and contact area information.

When considering column base plates, it is important to only consider the surface area which actually transfers load from the column to the slab. This area is dependent on the dimensions of the column, and the thickness and stiffness of the base plate. However, neglecting the effect of stiffness, a simplified effective base plate dimension is given as: $d_{eff} = d + 4t$, where d is the racking leg or column dimension and t is the thickness of the baseplate.

Even though TR34 does not clearly stipulate the definition of a line-load, the ACI (2010) suggests that a load be considered as a line-load when its width is less than 1/3 of the radius of relative stiffness, *l*, of the slab (see Section 2.2.3). Analysis of line and uniformly distributed loads is done on the basis of elastic slab behaviour and is discussed further in Section 2.4.1.

It is important to note that when two or more loads are spaced close together, they often have a combined effect which can be more significant than the sum of their individual effects. Bearing in mind that point loads are usually simplified as being circular, we consider a similar simplification for closely spaced loads. For loads with centres closer than twice the slab depth, *2h,* to each other, they can be considered to act as a single load. This load is assumed to have a surface area equal to the sum of the individual areas, expressed as circles, plus the area between them, as shown in Figure 2.4. This is assumed to be applicable to loads with magnitudes and dimensions which are either identical or different. Throughout this project, loads of this type are referred to as 'combined point loads', which is considered to be a type of single point load.

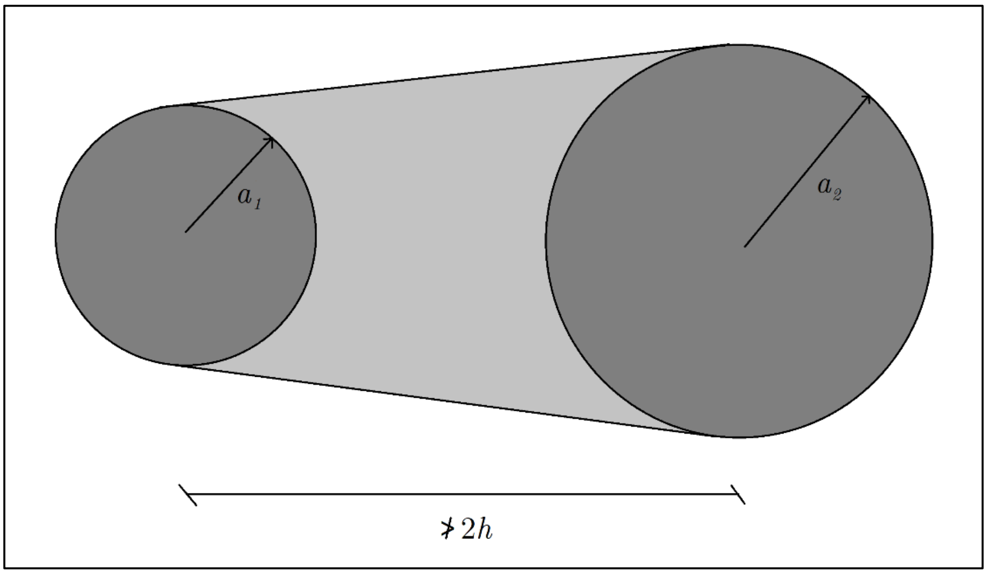


Figure 2.4: Equivalent contact area of two adjacent point loads with centres closer than twice the slab depth, 2h.

Bay regions – considering point loads

For the purpose of design, three point-load locations are considered based on the position of a load relative to slab edges and joints. These three positions are given below. The definitions given by TR34 have been adapted slightly to suit the objectives of this study. In the definitions below, *a* refers to the equivalent radius of contact area of the load (see Equation 1), and *l* refers to the radius of relative stiffness of the bay (see Equation 3).

Internal: the centre of the load is at a distance greater than $a+l$ from an edge.

Edge: the centre of the load is immediately adjacent to a free edge or joint and more than *l* from a corner, i.e. a free corner, the intersection between a joint and a free edge or the intersection of two joints.

Corner: the centre of the load is at a distance less than or equal to *l* from two edges or joints forming a corner.

Figure 2.5 shows the three possible point-load positions, as well as the measurements: *a* and *l*. It should be noted that, even though load transfer at joints is facilitated by dowels, aggregate interlock and/or other mechanisms, loads at edges adjacent to joints are considered in the same way as those at true slab edges.

The dimensions given for the three possible point-load positions can be used to divide any given bay into a combination of three types of regions/zones, namely: one or more internal region(s), multiple edge regions and corner regions. This is important, as the three types of regions will have different point-load capacities. Practical use of these regions will be demonstrated at a later stage.

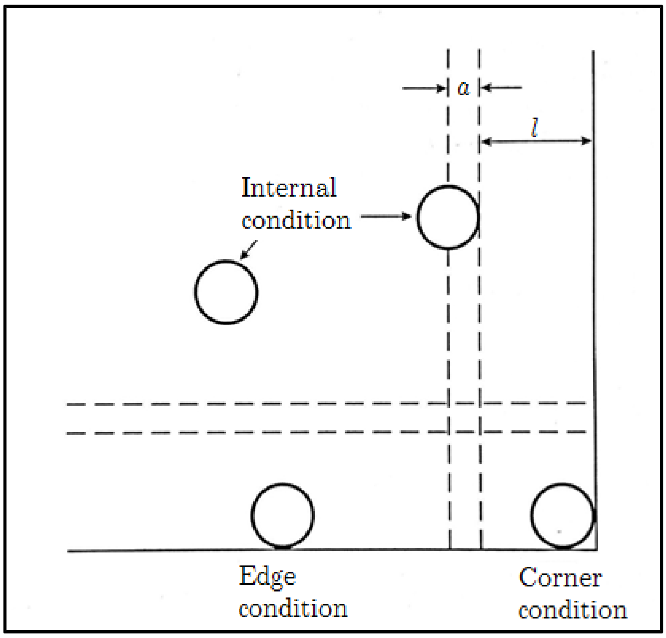


Figure 2.5: Point-load location definitions (Concrete Society 2013)

Bay regions – considering line loads

Similar to the point-load locations discussed above, line loads can also act at three different positions on a bay. These positions are established using a value known as the "characteristic" of the bay, λ [m^{-1}], calculated using:

$$\lambda = \sqrt[4]{\frac{3k}{Eh^3}} \qquad \text{Eq. (2)}$$

where:

k = modulus of subgrade reaction [$N/mm^2/mm$]

E = short-term modulus of elasticity of concrete [N/mm^2]

h = slab thickness [mm]

The three bay regions where a line load can act are as follows:

Internal: the load acts at a distance of at least $3/\lambda$ [m] from an edge.

Middle: the load acts at a distance of at least $1/\lambda$ [m], but less than $3/\lambda$ [m] from an edge.

Edge: the load acts at a distance less than $1/\lambda$ [m] from an edge.

These geometries can be used to divide a bay into another combination of three types of regions, independent of the bay regions considering point-loads, namely: one or more internal region(s), middle region(s) and edge region(s). Figure 2.6 shows these three bay regions for an arbitrarily orientated line-load. As is shown, a line load can act across multiple regions. It is therefore convenient to divide certain line-loads into segments, and consider each segment individually, based on the region where it is located.

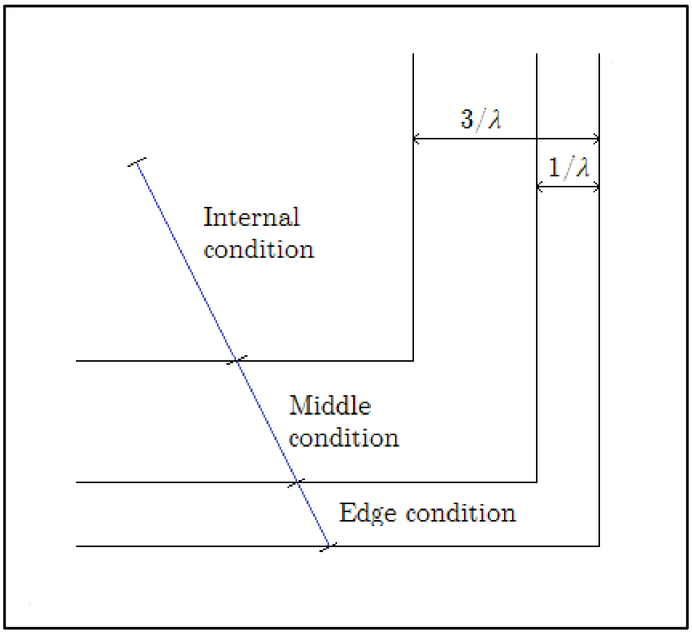


Figure 2.6: Line-load location definitions

Bay regions – considering uniform distributed loads (UDLs)

Since the effect of a UDL on a bay is independent of the position of the load, it is unnecessary to define bay regions as for point and line loads.

Loading patterns

The pattern in which loads act often has a major effect on their impact on a slab. For static loads, patterns often emerge in which two or more loads need to be considered as a unit, as discussed previously. For example, back-to-back pallet racking loads, as shown in Figure 2.2, often govern the slab design process.

Other racking loads, which often form influential loading patterns, include (Concrete Society 2013):

a. Mobile pallet racking with loads applied to rails. Either point or line loads.

b. Live storage systems

c. Drive-in racking

d. Push-back racking systems

e. Cantilever racks

f. Clad rack structures

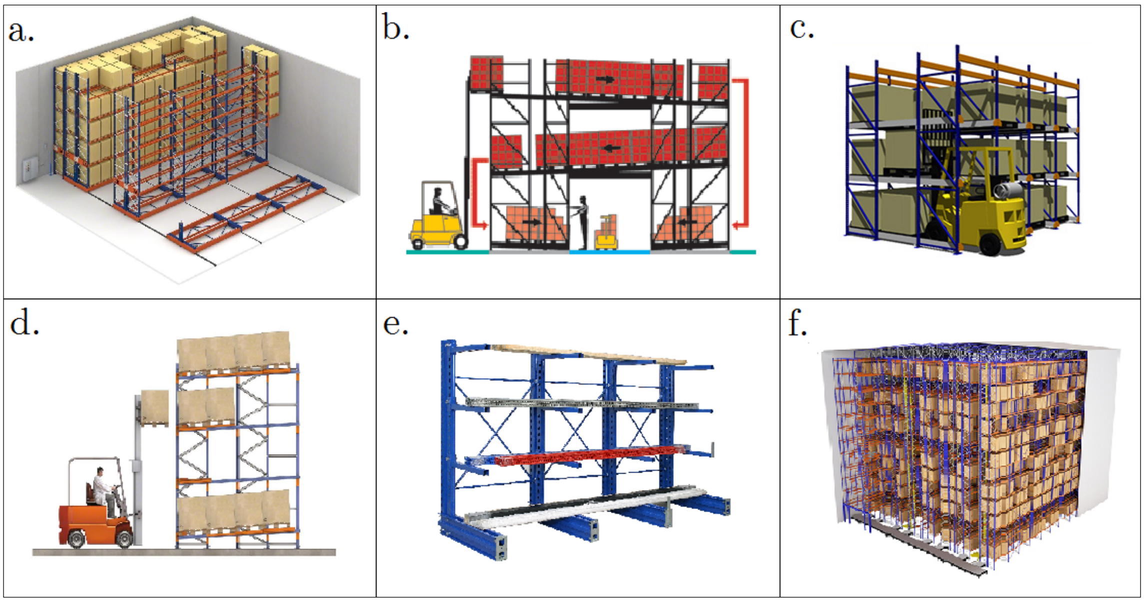


Figure 2.7: Various industrial racking systems:
a. Mobile pallet racking (Mecalux 2017a)
b. Live storage systems (Nilkamal 2017)
c. Drive-in racking (Tranpak 2017)
d. Push-back racking systems (Mecalux 2017b)
e. Cantilever racks (Krost shelving and racking 2017)
f. Clad rack structures (AR racking 2017)

Considering dynamic loads, the pattern in which they move are sometimes crucial. Heavy MHE can have detrimental effects on a slab, if proper measures aren't taken to ensure adequate reinforcement is placed at regularly trafficked points. Determination of the loading which a specific point on a slab is subjected to, due to dynamic loads, is a statistical process which is considered to be outside the scope of this study. However, the ability to identify the following movement zones is an important step in providing appropriate reinforcement layouts.

Free movement zone: An area where the movement of MHE and persons are considered to be completely random.

Defined movement zone: An area where MHE and other dynamic loads typically move in specific patterns, regularly imposing loads on points along certain pathways. Knowledge of such areas is important in the design phase of slabs.

2.2.3 Radius of relative stiffness

Westergaard (1925, 1926) first introduced the concept of a radius of relative stiffness, denoted by '*l*' (Concrete Society 2013). Since then, the term has often been used in research regarding slabs and is an important value in slab design and analysis, as it conveniently quantifies the stiffness of a slab. Stiffness, in this case, is seen as the resistance to deformation of a slab. Calculation of the radius of relative stiffness is done as follows:

$$l = \left[\frac{E \cdot h^3}{12 \cdot (1 - \nu^2) \cdot k}\right]^{0.25} \qquad \text{Eq. (3)}$$

where:

$E =$ short-term modulus of elasticity of concrete [N/mm^2]

$h =$ slab thickness [mm]

$k =$ modulus of subgrade reaction [N/mm^2/mm]

$\nu =$ Poisson's ratio, taken as 0.2

It is easily seen that, since: $(1 - 0.2^2) = 0.96$, ν will have little influence on l. Similarly, we see that the more compressible the soil is and the deeper the slab, the larger the value of l will be.

2.2.4 Soils and support structures underneath SOG

Soil factors

For any ground-supported structure, due consideration of the geotechnical factors at play is vital. Advanced geotechnics allow in depth analysis of a multitude of soil characteristics which might affect a structure at some point in its lifetime.

For this project, however, only the three characteristics described in the following paragraphs are deemed relevant when considering the design approach outlined by TR34 (Concrete Society 2013). Numerical values which represent the characteristics will usually require the involvement of a qualified geotechnical scientist/engineer. These values are then incorporated into the slab-design process to account for the behaviour of the underlying soil layers.

Firstly, the presence of long-term settlement is a factor which could have unfavourable effects on any structure, including slabs-on-ground. Early detection of this phenomenon will most likely mean that some mechanical work must first be done on the soil, before construction commences.

Next, we consider the carrying/bearing capacity of the soil. As the name suggests, this is the capacity of the soil to support loads which are applied to it and can be quantified as the maximum average pressure applied to the soil which does not result in shear failure of the soil.

Lastly, and possibly most importantly: the modulus of subgrade reaction, denoted by '*k*'. This single value for a specific region of soil is used directly in a multitude of widely accepted slab design equations. According to theories by Westergaard (1923, 1925, 1926) and Winkler (1867), slabs-on-ground rest on an ideal subgrade which acts as a vertical linear spring at all points, with '*k*' a proportionality constant. It represents the reaction force of a soil, similar to a spring constant, when compressively displaced a unit distance. This constant has units of pressure (kPa) per unit deformation (m), which is often written as kN/m^3.

The influence of soil moisture content on subgrade reaction should be noted. The extent of this influence depends on the soil texture, density and the activity of clay minerals present. Generally, increased moisture results in decreased supporting strength, indicating that adequate surface and subgrade drainage is crucial (ACI 2010).

Values of k can be verified in accordance with standardised plate bearing tests, for example, Eurocode 7 (British standards institution 2004b, 2007).

Calculation of the required slab thickness is not significantly influenced by small changes in k. Therefore, knowledge of the exact k value is typically not essential (ACI 2010).

It has been suggested (Concrete Society 2013) that the k-modulus of a soil can be enhanced through the addition of a granular subbase. However, seeing as this does not typically affect thickness design for flexural stresses, TR34 recommends design using the k value of the subgrade without any modifications.

Soil layers

Most ground soils consist of various layers, and appropriate knowledge of these layers serves as a valuable tool when dealing with ground supported structures, even though they are often simplified.

Sub-base: This is the top portion of soil and is in full contact with the slab, or base slab, if present. The three main purposes of the sub-base, as defined by TR34 (Concrete Society 2013), are to:

- Transmit loads on the slab to the subgrade.
- Provide good quality support to the slab.
- Provide a level platform for constructing the slab
- Provide suitable stability to accommodate construction activity.

Sub-bases typically consist of well-graded granular material with a minimum compacted thickness of 150mm.

Subgrade: This layer has no direct contact with the slab and may be made up of natural ground, imported fill or stabilised or dynamically compacted in-situ soil. It should provide uniform support without hard or soft spots.

Densification of the sub-base and subgrade layers is often performed, by means of mechanical compaction, in order to improve the reactive properties of the soil, including its k-modulus (ACI 2010).

In order to achieve certain specialised slab functionalities, additional layers are occasionally placed underneath a slab (see Figure 2.8). These layers might include:

Membranes:	Normally made of some form of plastic. A membrane reduces the friction between the slab and the sub-base and impedes the loss of water or fines from the slab concrete to underlying layers.
Insulation layer:	This device creates a heat transfer resistant barrier, valuable for temperature controlled rooms. Heating mats could also be present.

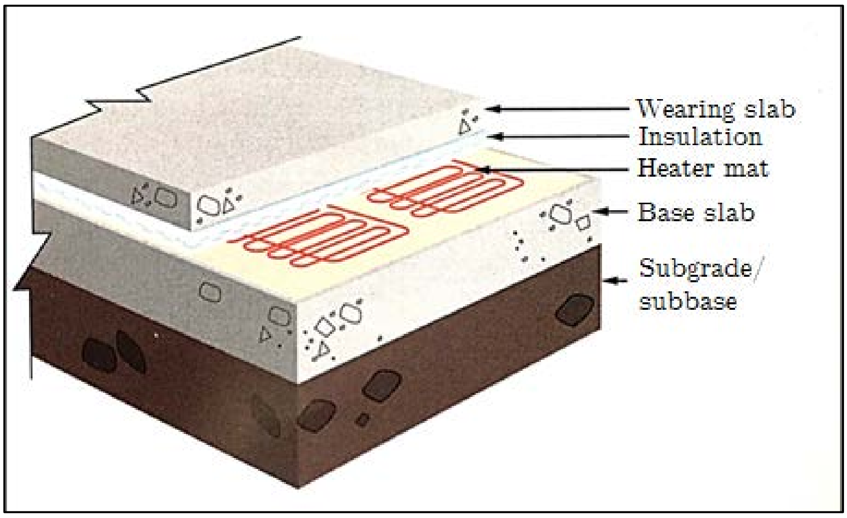


Figure 2.8: Illustration of possible SOG layers (Concrete Society 2013)

Piles:

Piles can be used in ground-supported slabs, however it is not typically complimented by the use of fibre-reinforced concrete. Therefore, it is not discussed further.

2.2.5 Bending moments for internal loads

In order to illustrate the effect which a point load has on a ground supported slab, as well as the effect of multiple point loads acting on the slab within an influential distance of each other, TR34 (Concrete Society 2013) provides the following explanation, aided by Figure 2.9.

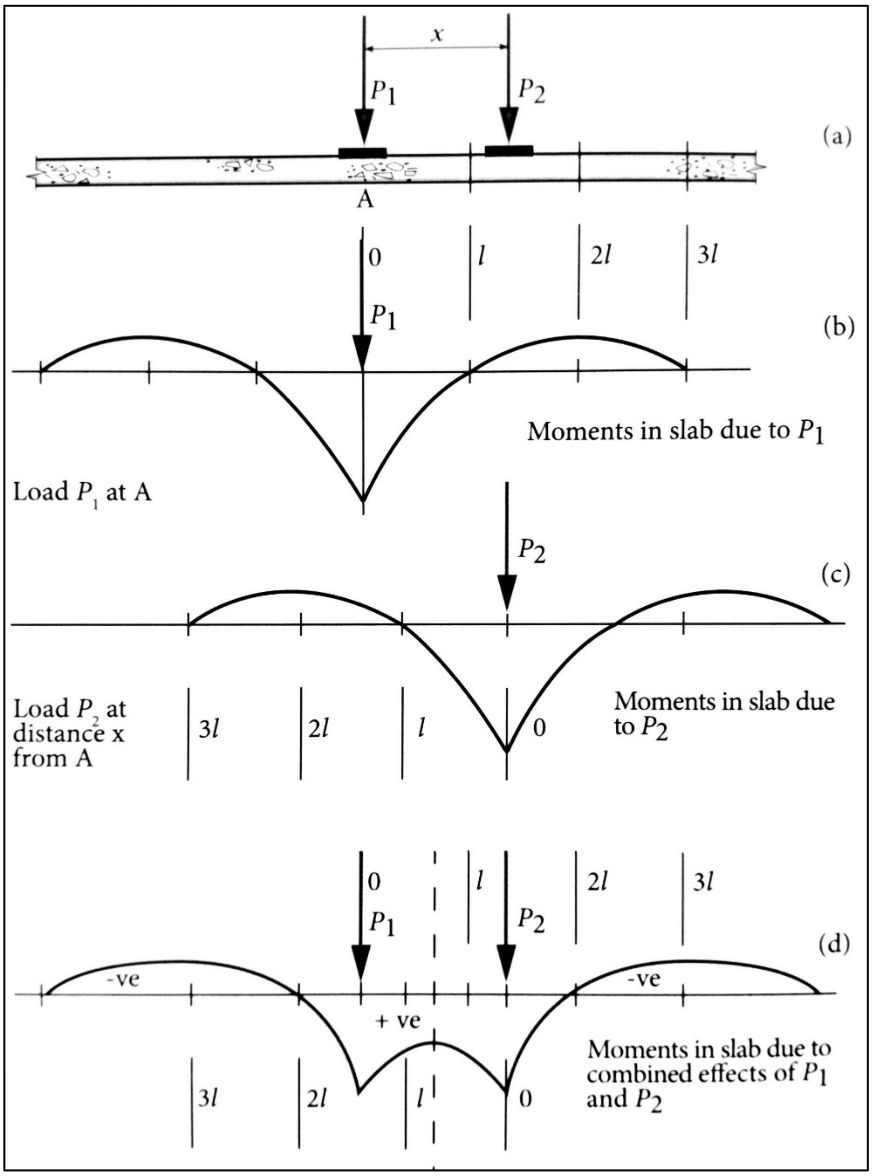


Figure 2.9: Schematic of distribution of elastic bending moments for internal loads (Concrete Society 2013):
(a) Typical load case; (b) for load P_1;
(c) For load P_2; (d) for the combined loads P_1 and P_2.

Considering a single point load P_1, the maximum positive bending moment within a SOG will occur directly beneath the load at point A. As the distance from the load increases, the circumferential moment decreases and reaches zero at a distance 1.0*l* from the load (see Section 2.2.3). From there the bending moment becomes negative and gradually increases in magnitude. At a distance of 2.0*l* from the load, the maximum negative bending moment is reached, which is significantly less than the maximum positive bending moment. After this second peak, the bending moment again approaches zero at $3.0l$ from the load.

The influence of an additional load P_2 at any distance x from A is as follows:

- If $x < l$, the positive bending moment at A will increase
- If $l < x < 3l$, the positive bending moment at A will decrease slightly
- If $x > 3l$, the effect of the additional load on the bending moment at A will be negligible.
- If $2l < x < 6l$, the additional load will increase the negative bending moment

The concepts of circumferential and radial moments are illustrated in Figure 2.10, which shows the case of a single point load applied to the interior of a large concrete SOG.

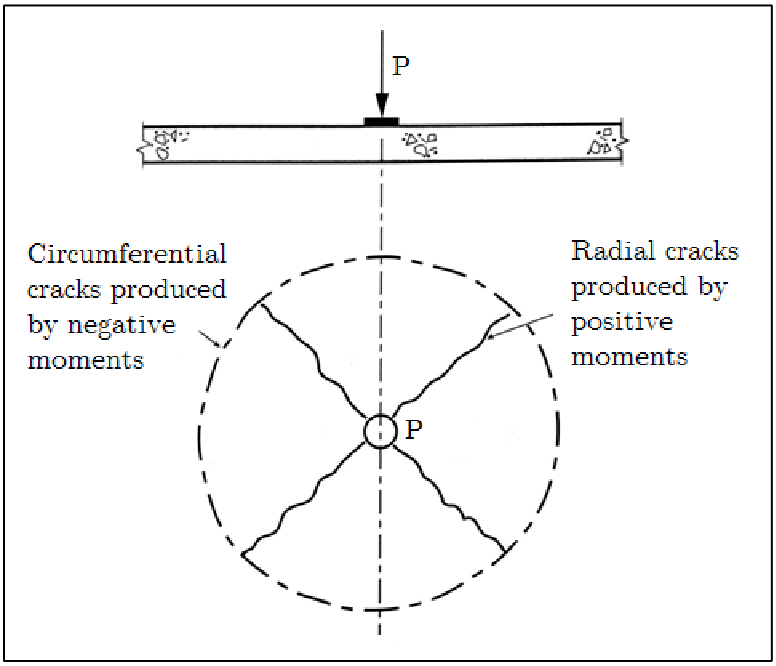


Figure 2.10: Development of radial and circumferential cracks in a concrete ground-supported slab (Concrete Society 2013)

As the load increases, the flexural stresses below the load will become equal to the flexural strength of the concrete. At this point, yielding of the slab will start, resulting in radial tension cracking of the bottom of the slab. This is caused by positive tangential moments and is commonly known as a fan-mechanism.

Subsequent to this yielding, and with further load increase, moment redistribution is assumed to occur, which prevents further increase of positive moments beneath the slab. However, a substantial increase in circumferential moments, some distance away from the load, is expected. Tensile cracking will occur in the top of the slab when the maximum negative circumferential moment exceeds the negative moment capacity of the slab, i.e. the moment capacity of a plain concrete section. When this condition is reached, failure is considered to have occurred as the design criterion of SOG is to avoid surface cracks.

Since two single point loads at a distance less than $3l$ from each other, will have an effect on the slab which differs from the effect which they would have if they acted independently, definition of a new type of point load is warranted. The term "dual point load" is therefore used to describe a pair of single point loads which are spaced a distance, x, from each other, where x is less than $3l$, but more than $2h$ (see Figure 2.4). By the same logic, the term "quadruple point load" refers to two pairs of single point loads, or one pair of dual point loads, which act at a distance, y, less than $3l$ from each other. The spacing and expected failure patterns for these two load types are shown in Figure 2.11.

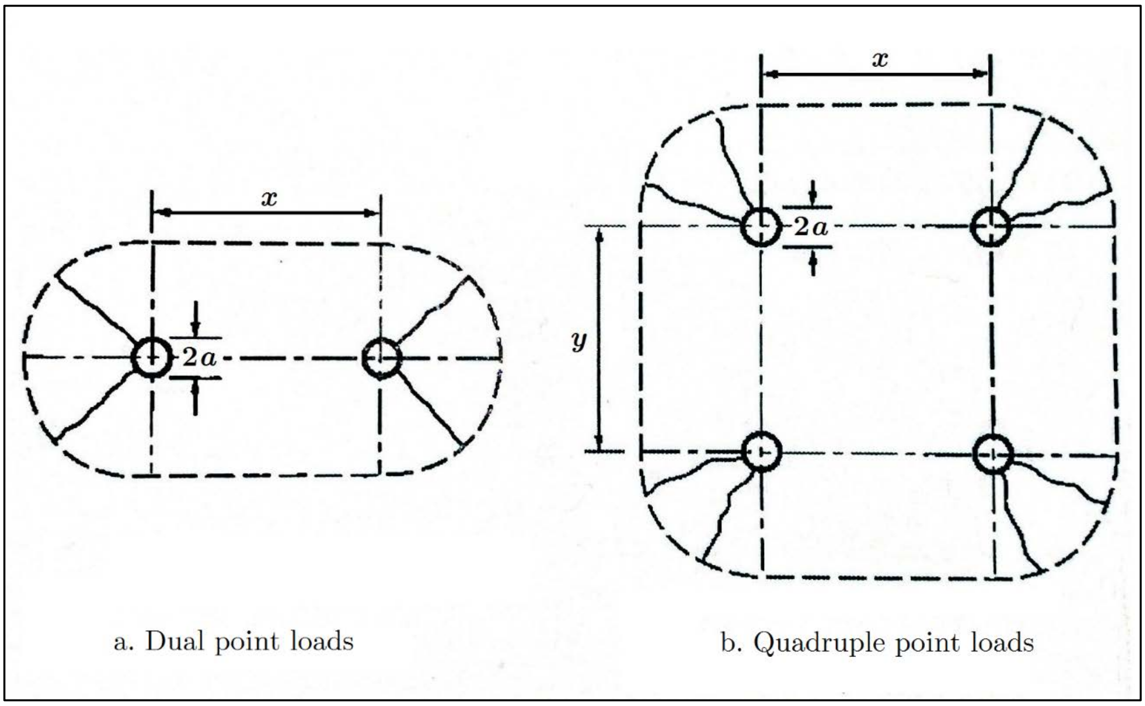


Figure 2.11: Spacing and expected failure patterns for dual and quadruple point loads (Concrete Society 2013)

Meyerhof (1962) used an ultimate strength analysis of slabs based on plastic analysis/ yield line theory and derived design formulae for the three main single load conditions: internal, edge and corner loads. He also considered combinations of two and four loads. Selection of the appropriate design formula is based on the load location, number of loads acting together and the size, *a*, of the load, relative to the radius of relative stiffness, *l*, of the slab. These formulae are often used by the algorithms developed in Chapter 3.

2.2.6 Joint layout and types

Due to various practical limitations when constructing concrete slabs, the need for joints at regular intervals is present in the vast majority of structures. These limitations include the fact that only a finite surface area of concrete can be placed in a certain session/workday. Furthermore, stress relief at joints is often vital in the prevention of shrinkage cracking. Joint spacing will directly influence the amount and character of random cracking that occurs (ACI 2010). Thus, where possible, joints should always be placed to optimise the positioning of static loads on the slab and to minimise the risk of cracking. See TR34 clause 11.10 for further guidance on joint layouts. Armouring and sealing of joints can also be done to improve joint performance and appearance.

A multitude of construction methods are industrially used; however, their specifications are outside the scope of this study.

Depending on several aspects of the desired slab performance, TR34 (Concrete Society 2013) provides information on the following joint types which are found in practice:

Free-movement joints: Designed to provide minimal horizontal restraint to movement, resulting from shrinkage and/or expansion. Vertical restraint and load transfer is accomplished by dowels or similar mechanisms.

Restrained-movement joints: Used in fabric reinforced floors. Reinforcing is continuous across the joint to allow shrinkage-induced stress relief at predetermined positions, elsewhere on the slab.

Sawn joints:	Joints formed by cutting a shallow groove in the top of the slab. This induces the formation of a crack underneath the cut. In spite of the presence of the crack, aggregate interlock provides some shear load transfer across the joint.
Formed joints:	A smooth joint is formed by allowing the concrete to set in a mould before casting the concrete on the opposite side of the joint.
Tied joints:	Sometimes provided to facilitate a break in construction at a point other than a free-movement joint.
Isolation joints:	Used to avoid harmful interaction of the slab with fixed elements at its edges or within its surface area, such as columns, walls or machinery bases. The ACI (2010) also recommends placing isolation joints at junctions with points of restraint, such as drains, manholes, sumps and stairways. Connecting the slab to any other part of the structure should be avoided whenever possible.

2.3 Fibre reinforced concrete

In the field of structural engineering, the use of fibres to strengthen concrete has become popular, due to various reasons. In this section an overview is given of the fundamental concepts involved when using fibres to reinforce concrete. Considering the scope of this study, some emphasis is placed on examining the production, use and properties of synthetic-fibre reinforced concrete (SynFRC).

2.3.1 Historical background

The use of fibres to strengthen cement-type structural members is hardly a new practice. Urban and rural establishments have utilised straw and horsehair to add structural integrity to huts and other structures for centuries. Even though the concept of fibre-reinforced concrete relies on these same basic principles, extensive research has helped to identify the causes and effects of various FRC features and has, over several years of trial and error construction coupled with academic research, helped to create materials with mechanical properties ideally suited to structural engineering applications (Walraven 2009, Ferrara, Meda 2006).

In spite of the long time period, for which fibres have been in use in structural engineering and construction, design of concrete structures implementing the use of macro synthetic fibres is still in its infancy. No universally accepted design methods are available (Concrete Society 2007) and no international building codes exist for FRC structures, limiting the potential use of fibre reinforcement in concrete construction (Di Prisco, Plizzari et al. 2009).

Further expansion of FRC methods and theory is crucial if an effective design method is to be developed. Examples of recent techniques which display such expansion include the use of Hybrid fibre systems (HyFRC) (Di Prisco, Plizzari et al. 2009). This involves combination of macro and micro fibres to create a more versatile composite. Combinations of fibres with steel fabric sheeting has been used in a similar manner (Concrete Society 2007, Banthia, Sappakittipakorn 2007).

2.3.2 Basic concept

In order to employ the use of fibres in any structural concrete, it is vital to grasp the basic concept on which FRC operates. Although counter intuitive, it is known that fibres typically do not have an active influence on increasing the flexural or tensile strength of concrete, similar to conventional steel. This is because the fibre reinforcing only comes into action after cracking of the concrete has begun and the fibres are stressed in tension (Concrete Society 2013). This behaviour is caused by fibre bridging mechanisms across crack surfaces (Buratti, Mazzotti et al. 2011) and is described as influencing the post-cracking behaviour of the concrete.

2.3.3 Fibre types currently in use

A wide variety of different types of fibres display effective strengthening of concrete. The three most commonly used fibres are steel, glass and synthetic fibres, typically made of polymers. The ability to manipulate these polymers on a microscopic level has facilitated micro-fibres, typically only a few microns in length, being used in combination with macro-fibres, typically 30-75mm in length.

Apart from tensile capacity, bond strength has been shown to be an important factor influencing the effectiveness of fibres (Solyom, Balazs 2016). Alberti et al. (2016) studied the pull-out behaviour of polyolefin fibres and noted the effect of inclination, embedded lengths and matrix type on the performance of fibre reinforcing. According to TR65 (Concrete Society 2007), mechanical deformations and other surface preparations can improve bond strength significantly, up to the point that fibre rupture occurs.

Considering the specific objectives and scope of this study, specific consideration of polypropylene and polyethylene fibres is given as follows.

<u>Physical and mechanical properties of polypropylene and polyethylene fibres in concrete</u>

Tensile strength:	Due to the general characteristics of polymers, high tensile strength is achieved at low weight percentages (Concrete Society 2007).
Bonding with concrete:	Bonding is purely mechanical with no chemical reaction taking place (Concrete Society 2007). This can be ascribed to the hydrophobic nature of these fibres (Babafemi, Boshoff 2017). Mechanical agitation could be beneficial to fibre anchorage (James, Gopalaratnam et al. 2002).
Concrete compressibility:	At typical fibre volumes, no effect on the compressive strength of the concrete is expected (Di Prisco, Plizzari et al. 2009). However, tests have revealed a more ductile failure mode of concrete, subsequent to fibre addition. Increasing fibre content without adjusting aggregate proportions can also cause poor workability, more bleeding and segregation, more entrapped air and lower unit weight. This causes lower compressive strength (James, Gopalaratnam et al. 2002).

Relative popularity:	According to a market trend analysis and literature survey conducted by Alani and Beckett (2013), polypropylene fibres are the most sustainable and desirable, when compared to other synthetic fibres used in FRC.
Comparison with SFRC:	Alani and Beckett (2013) also showed that there is a favourable comparison between concrete reinforced with synthetic fibres, at a dosage of 7 kg/m^3, and hook end steel FRC, at a dosage of 40 kg/m^3.

2.3.4 Testing the effects of fibres on concrete strength

TR34 outlines the testing of a FRC beam sample according to EN 14651 (British standards institution 2005). This method is known as flexural testing and is widely preferred for quantifying FRC performance (Concrete Society 2007). Beam tests often display high variability. Thus, due consideration should be given to testing consistency and determining the characteristic values.

Specimens 150mm wide by 150mm deep, on a span of 500mm, are tested under central point loading. A 25mm deep notch is saw cut into the side face of each specimen, as cast. The notched face is then placed in tension at the bottom of the test specimen. During testing, the crack mouth opening displacement (CMOD), i.e. the increase in width of the notch, is measured and the load, *f*, recorded at CMODs of 0.5, 1.5, 2.5 and 3.5mm. Alternatively, the central deflection can be measured and the load recorded at deflections of 0.47, 1.32, 2.17 and 3.02mm. A test set should consist of at least 12 samples. A typical graph of applied load, F_R, against CMOD is shown in Figure 2.12.

Each load is used to derive a 'residual flexural tensile strength' f_R, as follows:

$$f_R = \frac{3 \cdot F_R \cdot s}{2 \cdot b \cdot {h_{sp}}^2} \qquad \text{Eq. (4)}$$

where:

$F_R =$ applied load at stage R [N]

$h_{sp} =$ depth of the section, to the notch tip (±125mm)

$s =$ the span (500mm) $b =$ the width (150mm)

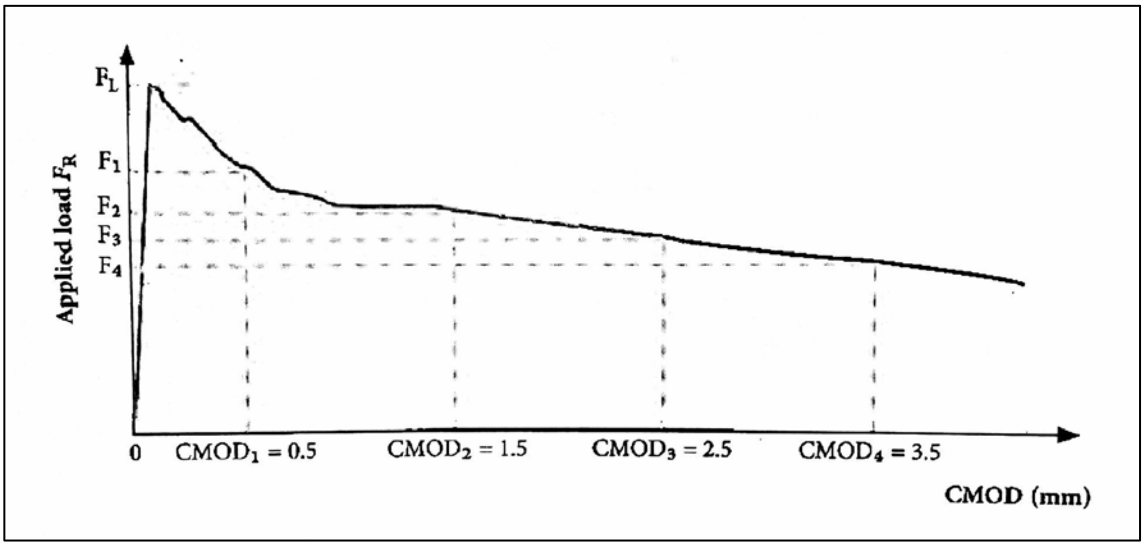


Figure 2.12: Typical graph of test load (F_R) vs. CMOD (Concrete Society 2013)

The four values: f_{R1}, f_{R2}, f_{R3} and f_{R4} are reported for each of the 12 samples and the characteristic value of each is used.

Then, implementing the method outlined by RILEM (2002), the axial tensile strengths, σ_{r1} and σ_{r4}, for two crack widths are considered, as follows:

$$\sigma_{r1} = 0.45 \cdot f_{R1} \qquad \text{Eq. (5)}$$

$$\sigma_{r4} = 0.37 \cdot f_{R4} \qquad \text{Eq. (6)}$$

where:

f_{R1} = the residual flexural strength at CMOD 0.5 mm

f_{R4} = the residual flexural strength at CMOD 3.5 mm

The crack depths are taken to be 0.66 and 0.90 of the beam depth. In the floor section, at ultimate limit state (ULS), it is assumed that the axial tensile strength at the tip of the crack is σ_{r1}, and at the tension face crack opening it is assumed to be σ_{r4}, with a triangular distribution between the two points.

The two values derived above are used in the design method discussed in Chapter 3. It should be noted that, regardless of the flexural strength values determined, the shear capacity of macro-fibre-reinforced concrete should be assumed to be equal to that of unreinforced concrete (Concrete Society 2007).

2.3.5 Production and use of SynFRC

Mixing and mix design

For the purpose of slabs-on-ground, significant adaptation of the design of the concrete mix is unnecessary when employing fibres for reinforcement. However, TR34 (Concrete Society 2013) recommends the following alterations to the concrete mix design, to accommodate the addition and effective use of fibres.

Firstly, an increase in fine aggregate content could be required to improve fibre dispersion and aid compaction and finishing of the FRC. The use of water-reducing admixtures could also be valuable as it will assist fibre dispersion throughout the mix while inhibiting drying shrinkage.

Furthermore, TR34 warns of the possibility of fibres agglomerating into balls when added to a concrete mix and advocates that steps be taken to counteract this.

TR65 (Concrete Society 2007) outlines suitable mixing practices to ensure optimised functionality of the fibres in concrete. When mixing SynFRC, fibres should be added to the mixer first, followed by rough aggregate to evenly distribute the fibres. The benefit of using ready-mixed concrete, mixed at the supplier plant and then transported to site, as compared to site-mixed concrete is mentioned as it typically involves more controlled conditions, and therefore yields more consistent concrete properties. The primary objectives of effective SynFRC mixing is identified as:

- Adequate fibre dispersion throughout the concrete
- Prevention of fibre balls that could cause pump blockages
- Ensuring that the fibres do not unduly affect the quality of the final concrete
- Even fibre distribution throughout the concrete mix.

It is recommended that the following practical considerations (Concrete Society 2007) be confronted before assigning the use of synthetic fibres in concrete:

- Ease of batching
- Ability of fibres to homogenously disperse through a given concrete and not ball up
- Effect on concrete consistence, slump and flow

- Effect on concrete pumpability

TR65 (Concrete Society 2007) lists some specific applications of SynFRC, as follows:

- Pavements and hardstandings
- Roads
- Structural screeds
- Domestic floors
- Agricultural applications

Other uses, also from TR65, include:

- Cast in situ concrete
  - Tunnel linings
  - Railways/non-magnetic applications
  - Marine/coastal applications
  - Walls
  - Water-retaining structures
- Precast concrete
  - Paving flags
  - Pipes and ancillary products
  - Cable troughs
  - Formwork for bridge decks
  - Piles
  - Staircase units

2.3.6 Properties of SynFRC

The use of synthetic fibres to strengthen concrete is often a viable option. However, in some cases, a different reinforcement alternative may be more suitable to the specific situation. This section outlines the properties of concrete reinforced by synthetic fibres (SynFRC) and describes some common advantages and disadvantages of its use.

Figure 2.13 shows typical flexural load vs. deflection curves of SynFRC with two to four percent fibre content by volume. It is shown that deflection varies linearly with flexural load up to the point of first cracking. Subsequent to cracking, transfer of loads to the fibres takes place. The load is then increased further, until the fibres also start breaking. As can be seen, the post-cracking load is sometimes higher than the load causing initial cracking (James, Gopalaratnam et al. 2002). However, according to TR65, the fundamental post-crack response will typically be strain softening. This occurs after the modulus of rupture (MOR) is reached and cracking occurs. If a strain hardening response is required, bar or fabric reinforcement would be more efficient (Concrete Society 2007).

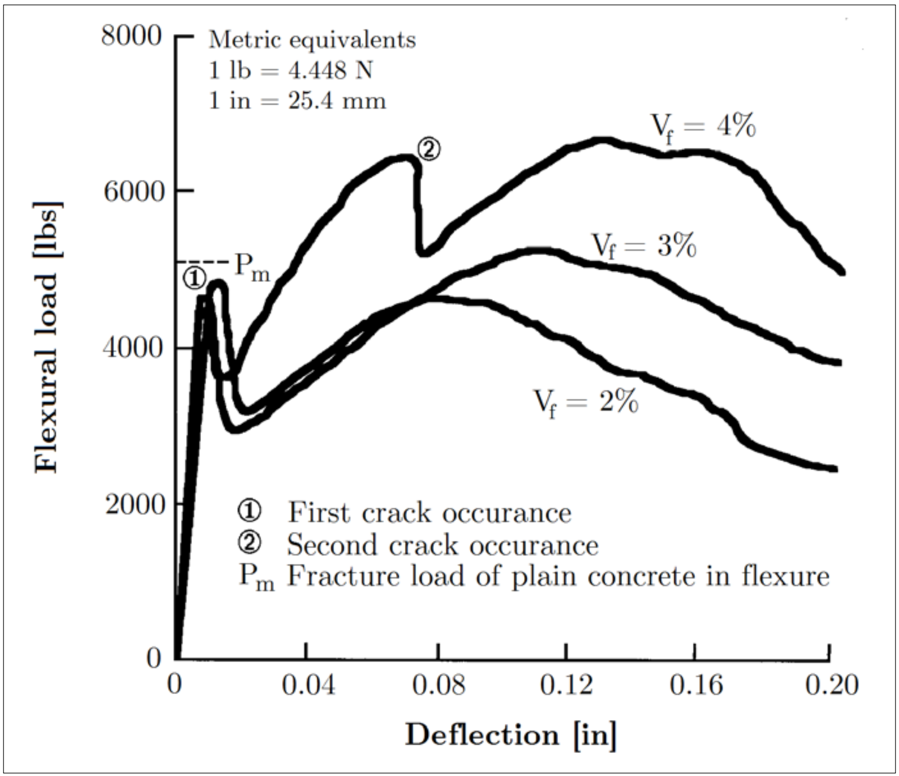


Figure 2.13: Typical flexural load-deflection curves of polyethylene FRC for various fibre contents (James, Gopalaratnam et al. 2002)

Long term loading of FRC structures has been shown to have an effect on fibre pull-out creep (Nieuwoudt, Boshoff 2017). However, concrete creep can be positively or negatively influenced by the presence of fibres. According to Zhao, Yu et al. (2016), fibres with elastic moduli much higher than plain concrete can clearly restrict creep, while fibres with lower elastic moduli have the opposite effect. Babafemi and Boshoff (2017) have shown that the rate of load application also has an effect on the tensile strength of FRC.

The effect which the use of SynFRC has on the cost of a project can be either positive or negative, depending on various factors, as described below.

Advantages of using SynFRC

Probably the most significant advantage and a vital trait of SynFRC, when compared to plain concrete, is its greater tensile strength and toughness. This refers to the energy absorbed by a test specimen as it is deformed (Concrete Society 2007), predominantly in the plastic state. This leads to a corresponding increase in flexural strength and toughness and is vital in providing increased post-cracking residual strength (see Figure 2.13). Excellent durability and impact resistance has also been noted, with high long-term stability performance resulting from the ability of the concrete to retain its toughness and strength over many years (Concrete Society 2007).

In structural FRC members, fibres are activated as soon as a crack in the concrete forms (Walraven 2009). The presence of synthetic fibres in concrete members therefore reduces crack widths, but increases the number of cracks forming, as loads can be transmitted across cracks.

Apart from purely structural considerations, the addition of synthetic fibres to concrete has also demonstrated superior corrosion resistance, compared to steel fibres, as well as lower susceptibility to harmful acid/base reactions. Construction time and complexity is also known to be less for SynFRC structures than for structures utilising steel, and transportation and handling is much more convenient.

From an environmental standpoint, SynFRC also demonstrates various benefits. Practices in the field of polymer science enable acceptable sustainability of synthetic fibre production and use. Although reuse of fibres in SynFRC is unlikely, the possible recycling and non-problematic disposal of unused fibres makes it an attractive building material. SynFRC with low fibre dosages could be broken up, crushed and reused. Also, no health and safety concerns to humans are prevalent.

Potential cost savings (Concrete Society 2007) are enabled by:

- No steel procurement
- Reduction of ancillary fittings
- Reduced labour costs
- Reduced mechanical and manual handling
- Reduced need for fixing and fabrication utilities
- Evasion of scrap, rework and returns

Disadvantages

One negative outcome of fibre reinforcing is that it has an adverse effect on the elastic properties of concrete, seeing as the post-cracking response of FRC is usually strain softening. Fibre use is also, in some cases, limited by economic considerations and the practicality of their introduction into the concrete (Concrete Society 2007).

As can be expected of polymeric materials, synthetic fibres are barely tolerant of extreme heat, eliminating its suitability in structures which might, in their lifetime, be exposed to fire.

Also, increases in project cost can result from the following factors (Concrete Society 2007):

- Changes to the concrete mix design
- Increased need for plasticiser/superplasticiser
- Higher handling charges from the concrete producer

2.3.7 Mechanical performance of MSFRC

In a recent study, Bester (2017) performed an in-depth experimental examination to investigate the effect of various factors on the mechanical performance of macro-synthetic fibre reinforced concrete (MSFRC), specifically in industrial flooring applications. Factors which were initially thought to influence the performance included: concrete compressive strength, coarse aggregate size, coarse aggregate volume, fibre dosage, and mixing time of the MSFRC. However, after completion of the study, it was concluded that fibre dosage is the only factor which has a significant influence on the residual flexural strength of MSFRC. This was statistically confirmed by simple- and multiple regression analyses. Based on this conclusion, equations were developed by which certain performance parameters can be predicted, based on the fibre dosage in the concrete.

Considering the scope of this study, two formulae from the above investigation are considered to be extremely useful, as they allow easy calculation of SOG bending-moment capacities, using known material properties and slab dimensions.

These two equations are as follows:

$$f_{R1,c} = 0.321 \cdot fD + 0.94 \qquad \text{Eq. (7)}$$

$$f_{R4,c} = 0.334 \cdot fD - 0.409 \qquad \text{Eq. (8)}$$

where:

$f_{R1,c}$ = Characteristic residual flexural tensile strength, corresponding to a crack mouth opening displacement (CMOD) of 0.5 mm [MPa]

$f_{R4,c}$ = Characteristic residual flexural tensile strength, corresponding to a CMOD of 3.5 mm [MPa]

fD = Fibre dosage of the concrete [kg/m^3]

These two equations are used to calculate the location of the neutral axis of a slab or bay in bending (see Section 3.6.1), as well as its bending moment capacity.

2.4 Slab-on-ground design approaches

2.4.1 Elastic slab design

Since effective performance of FRC requires at least some plastic cracking, elastic design of ground-supported slabs is somewhat conservative when fibre-reinforcing is employed, as the fibres will essentially increase the cracking strength of the concrete.

<u>Westergaard theory of elasticity and plasticity</u>

American Concrete Institute committee report 360R-10 (ACI 2010) states that theories relating to airport and highway pavements were the basis for many early design methods regarding slabs-on-ground. Westergaard (1923, 1925, 1926) investigated the behaviour of rigid pavements and developed a theory which is still in high regard today. This theory considers a homogenous, isotropic and elastic slab resting on an ideal subgrade that exerts, at all points, a vertical reactive force which is proportional to slab deflection. A subgrade which demonstrates this ideal behaviour is characterised as a "Winkler subgrade" (Winkler 1867). Experimental analysis has been carried out at the Arlington Virginia Experimental Farm and Iowa State Engineering Experiment Station, which showed agreement with Westergaard's theoretical slab analyses. An important condition, essential to these findings, is that the entire slab must remain in contact with the subgrade at all times (ACI 2010).

Regardless of the significance of the above theory, certain limitations mean that alternative methods of slab analysis need to be explored in order to consider more complex structures. For example, when making use of fibre reinforcement within the slab, the assumption of a plastic response at the bottom of the slab is essential to activating the fibres and considering their behaviour in response to certain loading conditions. Ground supported slabs tend to conform to the shape of the underlying subsoil as it deflects under loading (Concrete Society 2013). Thus, yield-line analysis (Section 2.4.2) is often considered to be a more accurate approach in analysing SOG.

Hetenyi (1971) carried out an elastic analysis of beams on elastic subgrade, which has been adopted as the basis of slab-on-grade design, particularly when considering line and uniformly distributed loads.

2.4.2 Yield-line theory of slab analysis

Kennedy and Goodchild (2004) compiled a detailed report covering the majority of concepts involved in yield-line theory and design. A brief summary of their work is given in this section, with specific reference being made to reinforced concrete slabs, in order to gain an understanding of the origin of methods and equations presented in Chapter 3.

As a result of the relative simplicity of this method, the use of computers is not always necessary for analysis and design.

Yield line theory revolves, to a certain extent, around typical failure patterns of structures and demands some knowledge and understanding of the subject. Design using yield line methods correlates with the ultimate limit state and therefore does not address serviceability issues. Formulae based on the yield moment can be used for deflection calculations.

When comparing different methods of design and detailing, yield line design gives the least weight reinforcement requirement and least complication (Goodchild 2000). This can be attributed to the fact that the designer is in full control of how moments are distributed throughout the slab and therefore has the opportunity to use simple reinforcement layouts. This results in a cost-effective solution for the designer, detailer, contractor and fixer, seeing as minimal materials are typically needed and complex reinforcing configurations are avoided.

Since the yield line design approach is conceptually different from elastic design approaches, it is helpful to consider the following basic concepts. Firstly, as previously mentioned, yield line design is a plastic method that investigates the post-yield/post-crack behaviour of a slab at ultimate loading. Thus, elastic deformations are neglected and adequate ductility to assume plastic behaviour is required (Concrete Society 2013). Slab weight is assumed to be negligible when compared to applied loads (Baumann, Weisgerber 1983). The yield-line method relies strongly on the concepts of moment redistribution and the formation of plastic hinges (ACI 2010).

The fundamental principle upon which the theory relies is that:

$$\boxed{\text{Work done in yield lines rotating} = \text{work done by moving loads}}$$

This is elaborated on, later in this section.

Kennedy and Goodchild (2004) define a yield-line as: “a crack in a reinforced concrete slab across which the reinforcing bars have yielded and along which plastic rotation occurs”. They further explain the yield-line theory as follows:

For illustration purposes, we consider a square slab, simply supported along all four its edges, as shown in Figure 2.14. This slab is subjected to a uniformly distributed load, which gradually increases until collapse occurs.

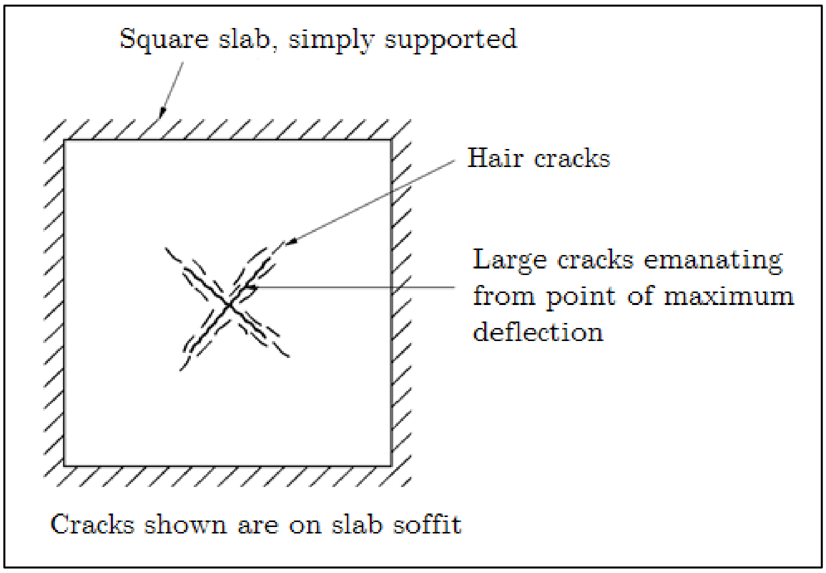


Figure 2.14: Onset of yielding of bottom reinforcement at point of maximum deflection in a simply supported two-way slab (Kennedy, Goodchild 2004)

Initially, at service load, the response of the slab is elastic with the maximum reinforcing stress and deflection occurring at the centre of the slab. At this stage, some hairline cracking will possibly occur at midspan, on the soffit where the flexural tensile capacity of the concrete has been exceeded.

Increasing the load accelerates the formation of these hairline cracks. Increasing the load further will increase the size of the cracks further and induce yielding of the reinforcement, initiating the formation of large cracks emanating from the point of maximum deflection.

On increasing the load yet further, these cracks migrate to the free edges of the slab at which time all the tensile reinforcement passing through a yield-line yields.

At this ultimate limit state, the slab fails. As illustrated by Figure 2.15, the slab is divided into rigid plane regions A, B, C and D. Yield lines form the boundaries between the rigid regions, and these regions, in effect, rotate about the yield lines. The regions also pivot about their axes of rotation, which usually lie along lines of support, causing supported loads to move.

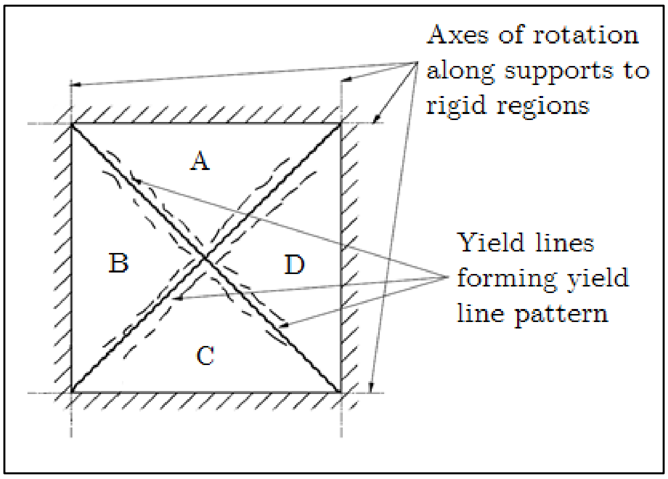


Figure 2.15: The formation of a mechanism in a simply supported two-way slab with the bottom steel having yielded along the yield lines (Kennedy, Goodchild 2004)

It is at this juncture that the work dissipated by the hinges in the yield lines rotating is equated to work expended by loads on the regions moving. This is what is referred to as Yield Line Theory.

Considering an infinite slab subjected to a single point load, it is intuitive that a slightly different failure pattern will occur from the one shown above. As described in Section 2.2.5, the slab will fail by a so-called “fan mechanism”, with positive radial yield lines and negative circumferential yield lines, as shown in Figure 2.16.

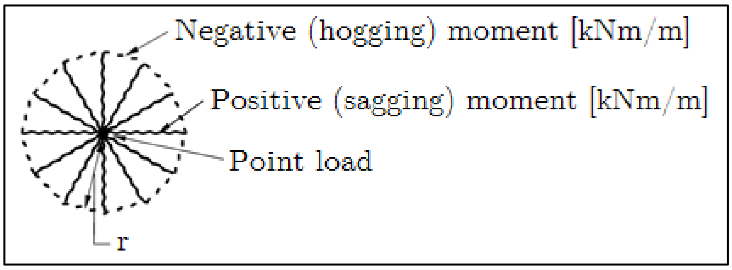


Figure 2.16: Fan collapse pattern for a heavy concentrated load onto a reinforced slab (Kennedy, Goodchild 2004)

The crack pattern of a slab subjected to edge and corner loads, respectively, has also been illustrated by Baumann and Weisgerber (1983) as shown in Figures 2.17 and 2.18.

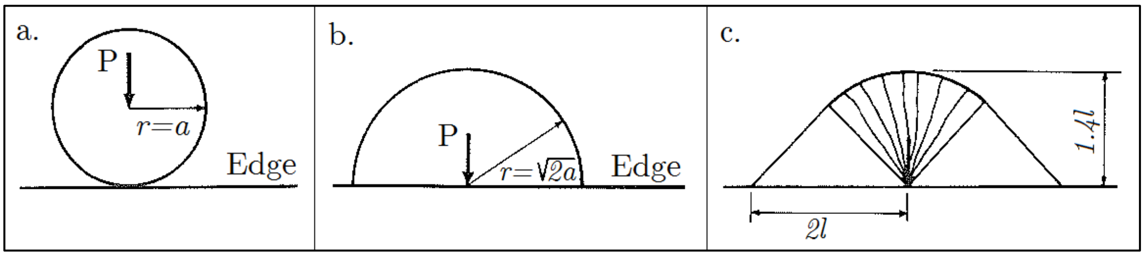


Figure 2.17: Load and crack patterns for an edge loaded slab (Baumann, Weisgerber 1983): (a) Circular load at edge of slab; (b) Semi-circular load at edge of slab; (c) Yield line formation due to edge loading

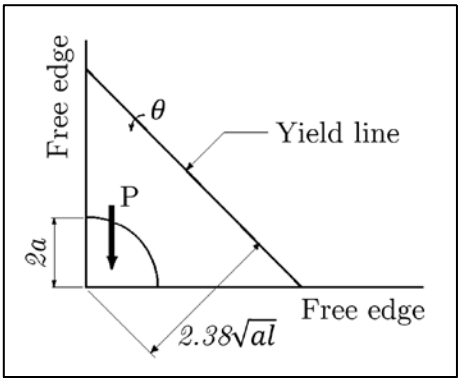


Figure 2.18: Quarter-circle loading and yield line pattern for corner loading (Baumann, Weisgerber 1983)

where:

a = equivalent radius of contact area of the load - see Equation 1 [mm]

l = radius of relative stiffness of the slab - see Equation 3 [mm]

With the above notion in mind, the fundamental yield line theory principle is elaborated on, by means of the so-called Work Method:

External energy expended by loads moving	=	Internal energy dissipated by rotations about yield lines
Expended	=	Dissipated
E	=	D
Σ (N x δ)$_{\text{for all regions}}$	=	Σ (m x l x ϑ)$_{\text{for all regions}}$

where:

N = load(s) acting within a particular region [kN]

δ = the vertical displacement of the load(s) N on each region expressed as a fraction of unity [m]

m = the moment in or moment of resistance of the slab per metre run [kNm/m]

l = the length of yield line or its projected length onto the axis of rotation for that region [m]

ϑ = the rotation of the region about its axis of rotation [m/m]

Although detailed derivations are not discussed, it is known that up-to-date methods of slab analysis, as used in this study, rely on equality of internal and external work (Baumann, Weisgerber 1983) as illustrated above. Mathematical formulae for slab analysis and collapse load determination, based on plastic theory, have been advocated by Meyerhof (1962) and were later developed further by Losberg (1978) and Baumann and Weisgerber (1983).

2.5 Using SynFRC in yield-line theory slab-on-ground design

As stated in Section 2.4.2, yield line theory was developed and implemented with generally effective slab design being a main objective. The fact that slabs-on-ground are supported by non-rigid, compressible soils means that they are just as susceptible to the formation of yield lines. Thus, since the same fundamental principles can be applied to slabs-on-ground, they can be analysed and designed using the same theory.

The use of fibre-reinforcement is also easily incorporated in yield line theory. However, more experimental investigation is needed for FRC than regular or steel-reinforced concrete. Since yield line theory involves itself with the moment resistance of slabs, a theoretical yield moment is needed for load capacity calculations. While the theoretical moment capacity can be calculated from material properties and section details for conventionally-reinforced concrete, FRC requires physical testing using sample beams to determine this characteristic value. This, however, is the only special consideration needed to incorporate FRC into design methods using yield line theory, and can be done using the test described in Section 2.3.4.

Chapter 3

Algorithm development

3.1 Introduction to algorithm development

As mentioned previously, creating a software prototype capable of analysing and designing FRC slabs-on-grade, mainly depends on the development of a suitable set of algorithms.

In order to develop such algorithms, a specialised approach to analysing and/or designing a ground supported slab is adopted. This approach involves individual consideration of the bay(s) which make up any given slab. This is warranted by the fact that load transfer mechanisms across slab joints, i.e. between multiple bays, is neglected during this study. Therefore, when analysing and/or designing a slab-on-ground, each bay is analysed and/or designed separately, only considering the loads which have an effect on the bay under consideration. Design/analysis information from all bays is then collectively considered, when analysing and/or designing the complete slab.

Figure 3.1 illustrates a slab-on-ground consisting of three bays. As is shown, each bay is subjected to a unique set of point-, line and/or uniform distributed load(s). When analysing the slab, the loads on each bay is compared to the relevant load capacity of the same bay. Similarly, when designing a slab, each bay is designed to a unique and independent thickness, fibre dosage or an optimal combination of the two, depending on the objective of the user. If, however, it is required to have a uniform thickness and/or fibre dosage across the entire slab, the limiting bay thickness and/or fibre dosage can be set to all bays.

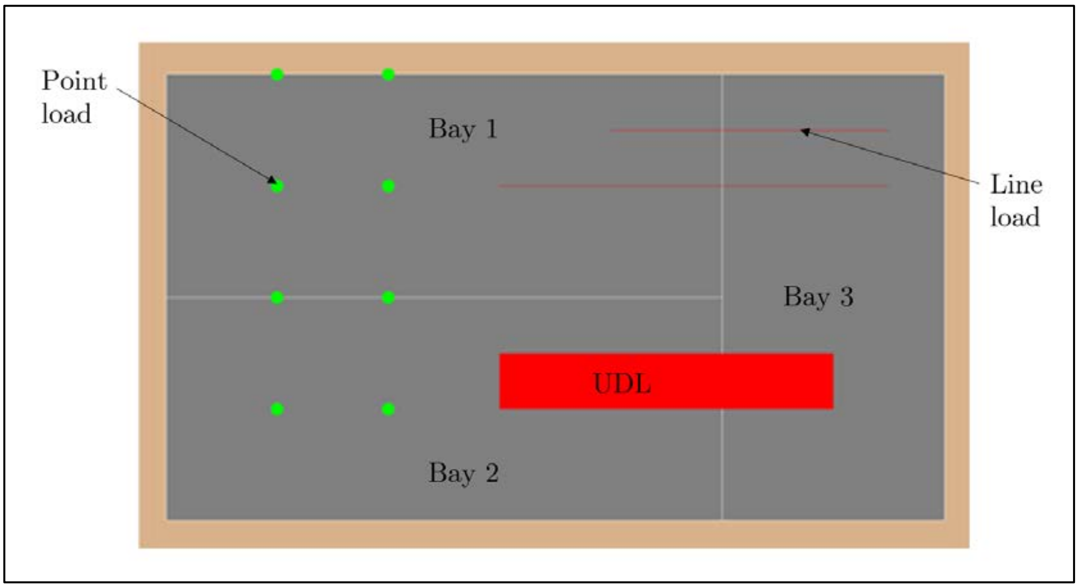


Figure 3.1: Plan view of a typical slab component setup

The complete slab-design algorithm is developed in three phases, each adding a new layer of functionality and utilising underlying layers and lower level algorithms. Therefore, the Design Algorithm will regularly employ the Analysis Algorithm, while the Optimisation Algorithm will regularly use the Design Algorithm. For the purpose of conceptual illustration, these layers are shown in Figure 3.2. A software model, incorporating the functionality of these algorithms is set out in Chapter 4.

The three respective phases are discussed in detail in Sections 3.3 to 3.5.

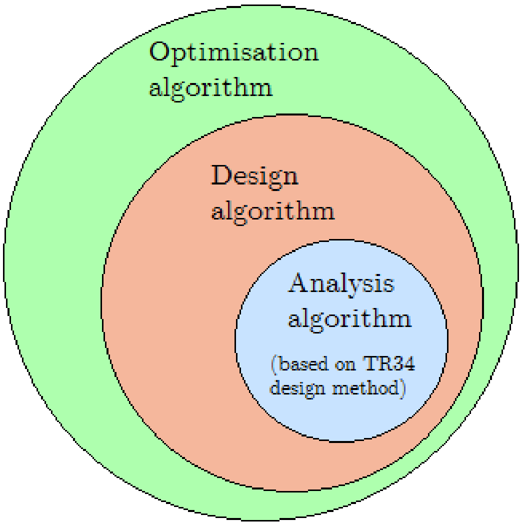


Figure 3.2: The three layers (phases) of the complete slab-design algorithm

3.2 Algorithm variables

When utilising the algorithms to perform a slab design, the user is required to provide certain characteristic values which are known or desired. The Analysis and Design algorithms can then be used to calculate any unknown values, compare load intensities to relevant slab capacity values and/or design a slab to ensure sufficient capacity. The Optimisation Algorithm can optimise the design by changing selected variables to minimise a cost function. As described in Sections 3.3 to 3.5, the following characteristics are considered as variables in the slab design process:

- The slab thickness
- The required minimum fibre dosage to be added to the concrete mix, according to the relevant fibre performance model (see Section 2.3.7).

Furthermore, the following parameters are also variable, but their values must be provided by the user, prior to slab analysis or design.

- Size and shape of slab bays
- The class of concrete being used
- Subbase and subgrade properties (modulus of subgrade reaction – k)
- Magnitude, dimension and location of all anticipated loads on the slab
- Regions on the slab which are expected to be subjected to MHE traffic

3.3 Analysis Algorithm

This is the basic algorithm deduced from the design guideline set out in TR34. This algorithm has the capability of analysing a given slab with all data such as thickness, fibre performance. and load information specified by the user. The algorithm will perform a predetermined sequence of calculations and comparisons to determine whether or not the slab has sufficient capacity to carry the loads placed on it.

The Design Algorithm (Section 3.4) will regularly utilise the Analysis Algorithm, to assess whether or not a given slab-setup is sufficient, and then make specific changes to parameters based on the results obtained.

3.3.1 Procedure for SOG analysis

The following series of steps outlines the process by which a slab can be analysed. Prior knowledge of the characteristic and geometric values of at least one bay is essential; however, there is no limit to the amount of loads which can be analysed. A visual illustration of the process is given in Appendix A, by means of a flowchart.

<u>For each point load:</u>

- Assign one or more bay(s) to the load, depending on where it is located. Multiple bays can be assigned, if the load is on a joint.
- Calculate the equivalent radius of contact area, a, of the load (see Equation 1).
- For each bay affected by the current point load:
  - Determine the radius of relative stiffness, l, of the bay (see Section 2.2.3).
  - Determine the neutral axis depth of the bay (see Section 3.6.1).
  - Calculate the shear face area of the load, depending on the bay-region where it is located (see Section 3.6.3).
  - Calculate the shear and bending moment capacities of the bay (see Sections 3.6.2 and 3.6.3). Also account for the region of the bay subjected to the loading: edge, corner or interior.
- Determine the capacity of the slab to carry the point load, based on the limiting capacity of the bay(s), in shear or bending.
- Compare the magnitude of the point load to the capacity of the slab to carry it. Identify all loads which exceed the relevant capacities. Also identify the loads which are closest to exceeding their capacities or which exceed their capacities by the greatest amount, i.e. the limiting point loads.

<u>For each line load:</u>

- Assign one or more bay(s) to the load, depending on the location of the line.
- Calculate the characteristic value, λ, of each bay affected by the load (see Equation 2).
- Divide the line load into segments (see Section 3.6.5)

- For each segment of the current line load:
  - Calculate the capacity of the slab to carry the segment, taking into account the bay and region where the segment is located and the proximity of the segment to the region boundaries (see Section 3.6.2).
  - Compare the most adverse load magnitude of the segment to the capacity of the slab to carry it. Identify all load segments which exceed the relevant capacities. Also identify the load segments which are closest to exceeding their capacities or which exceed their capacities by the greatest amount, i.e. the limiting line load segments.

<u>For each UDL:</u>

- Assign one or more bay(s) to the load, depending on the location of the load.
- For each bay affected by the current UDL:
  - Calculate the characteristic value, λ, of the bay (see Equation 2).
  - Calculate the UDL capacity of the bay, considering bending moments (see Section 3.6.2).
- Determine the capacity of the slab to carry each UDL, based on the limiting bay capacity.
- Compare the magnitude of each UDL to the capacity of the slab to carry it. Identify all loads which exceed the relevant capacities. Also identify the loads which are closest to exceeding their capacities or which exceed their capacities by the greatest amount, i.e. the limiting UDLs.

3.4 Design Algorithm

By regularly using the Analysis Algorithm, this layer will enable the determination of specific slab characteristics: thickness or fibre content, which will produce a successfully functioning slab, which is not over-designed. The inputs and outputs of this layer are not fixed and the Design Algorithm will have the capability of solving for either of these variables by using iterative solution techniques.

Depending on the objective at hand, this algorithm can be employed in one of three ways:

1. By providing bay dimensions, fibre dosages, characteristics and anticipated loads, use the algorithm to determine the required thickness(es).
2. By providing bay dimensions, thicknesses, characteristics and anticipated loads, use the algorithm to determine the required fibre dosage(s).
3. By providing bay dimensions, thicknesses, characteristics and anticipated loads, use the algorithm to determine the required f_{R1} and f_{R4} value(s).

The term "basic slab design" refers to the process of designing a slab in one of the three ways listed above.

3.4.1 Procedure for basic SOG design

As mentioned previously, basic slab design can be performed in one of three ways. These three types of basic design are deliberated in this section.

3.4.1.1 Basic slab design to suitable thicknesses

The design of a slab to a suitable thickness, or thicknesses, can be performed. This involves regularly employing the analysis procedure outlined in Section 3.3.1.

The relatively simple process of determining the suitable thickness value(s) requires all other variables, including fibre dosage, to be pre-specified by the user, and follows the following steps for each bay of a slab. A visual illustration of the process is given in Appendix B, by means of a flowchart.

1. Check if the pre-assigned fibre dosage of the bay is within the limits specified in Section 3.6.4. If not, adjust the fibre dosage to either the minimum - if the dosage is too low, or maximum - if the dosage is too high, allowable value.
2. Set the thickness of the bay to a starting value. Typically, the user will have entered an initial thickness value for the bay. If not, an arbitrary value of h = 200mm is used.
3. Determine, using the Analysis Algorithm, whether or not the bay has sufficient capacity to carry the loads which act on it, based on the current thickness and fibre-content of the bay.

4. If the bay capacity is exceeded by any load acting on it, the thickness of the bay must be increased. Therefore, increase the thickness by a calculated increment value. The increment value is based on the current thickness, the current load capacity of the bay considering the governing load type, *C*, and the magnitude of the governing load on the bay, *M*, and is calculated as:

$$increment_1 = h \times \left(\frac{M - C}{C}\right) \div 100 \qquad \text{Eq. (9)}$$

   After increasing the slab thickness, return to step (3.). Repeat this process until the load capacity of the bay exceeds the magnitude of the governing load.

5. If the bay capacity is not exceeded, its thickness can likely be reduced. This will ensure a more economical slab, which still has sufficient capacity to carry the applicable loads. Therefore, reduce the thickness by an increment value similar to the one calculated in Equation 9. However, seeing as the load capacity of the bay, *C,* exceeds the magnitude of the governing load, *M,* in this case, we rather calculate the increment value using:

$$increment_2 = h \times \left(\frac{C - M}{C}\right) \div 100 \qquad \text{Eq. (10)}$$

   After decreasing the slab thickness, return to step (3.). Repeat this process until the magnitude of the governing load exceeds the load capacity of the bay. Then increase the thickness by one increment, so that the capacity is sufficient, but not excessive.

6. If the fibre dosage of the bay was previously set to the minimum allowable value, calculate a new minimum allowable fibre dosage (Section 3.6.4) based on the post-design bay thickness. Set the fibre dosage of the bay to this new value.

3.4.1.2 Basic slab design to suitable fibre dosages

Determination of an acceptable fibre dosage is done by a process similar to the one for suitable thickness. However, there are several conceptual differences between bay thickness and fibre dosage, which warrant special consideration when designing a slab.

Firstly, considering the elastic analysis approach which is implemented for line-and uniform-distributed-loads (see Section 2.4.1), it is evident that the fibre dosage of a bay has no direct influence on its theoretical line-load and UDL capacities. This means that if the governing load on a certain bay is a line load or UDL, adjustment of the fibre dosage will be futile from a design perspective.

Secondly, even though the analysis of point loads acting on a bay considers the fibre content in bending moment capacity calculations, the same does not apply for calculation of shear capacity. This is due to the fact that macro-synthetic fibre reinforcing is assumed to have a negligible effect on the punching shear capacity of a bay (as mentioned in Section 2.3.4).

The following steps, to determine a suitable fibre-content for each bay, have been developed to take account of the considerations given. Similar to Section 3.4.1.1, this process requires all other variables: in this case including bay thickness and fibre type, to be pre-specified by the user. A visual illustration of the process is given in Appendix C, by means of a flowchart.

1. Check if the pre-assigned bay thickness value is acceptable, considering the minimum allowable slab thickness, specified by the user or TR34 (Concrete Society 2013). TR34 recommends a minimum design thickness of 150 mm for all slabs-on-grade. If not, set the bay thickness to the minimum allowable value.
2. Set the fibre-dosage of the bay to a starting value. Typically, the user will have entered an initial dosage value for the bay. If not, an arbitrary value is used which is the numerical average between the minimum and maximum allowable fibre dosages (see Section 3.6.4).
3. Determine, using the Analysis Algorithm, whether or not the bay has sufficient capacity to carry the loads which act on it, based on the current thickness and fibre-content of the bay.
4. If the capacity of the bay is exceeded by any load acting on it, it is possible that an increase in fibre-dosage will resolve this problem, depending on the type and nature of the governing load, as explained at the start of this section. Therefore, increase

the dosage by an increment value similar to the one calculated in Equation 9, however, considering fibre-dosage, *fD,* instead of thickness, as follows:

$$increment_3 = fD \times \left(\frac{M - C}{C}\right) \div 100 \qquad \text{Eq. (11)}$$

After increasing the dosage, check if the load capacity of the bay was increased. If the dosage increase had a positive effect, return to step (3.). Repeat this process until the load capacity of the bay exceeds the magnitude of the governing load.

If, however, the dosage increase had no effect on load capacity, any excess fibres should be removed to lower the cost of the bay. Decrease the fibre dosage by the same increment, until a decrease in load capacity is observed or until the minimum allowable fibre-dosage is reached, then increase the dosage by one increment so that fibre dosage does not limit the load capacity. If, at this stage, the load capacity is insufficient, the user will be advised to rather perform a basic design to a suitable thickness, as further adjustment of the fibre-dosage will not result in sufficient load capacity being obtained.

5. If the bay capacity is not exceeded, its fibre-dosage can likely be reduced. This will ensure a more economical slab, which still has sufficient capacity to carry the applicable loads. Therefore, reduce the dosage by the increment value calculated by:

$$increment_4 = fD \times \left(\frac{C - M}{C}\right) \div 100 \qquad \text{Eq. (12)}$$

After decreasing the fibre-dosage, return to step (3.). Repeat the reduction process until the magnitude of the governing load exceeds the load capacity of the bay. Then increase the thickness by one increment, so that the capacity is sufficient, but not excessive. If the load capacity of the bay is not affected by the reduction in fibre-dosage, the dosage is reduced until it is set to the minimum allowable value.

3.4.1.3 Basic slab design to suitable f_{R1} & f_{R4} combinations

If the fibre product to be used in a specific SOG design is unknown, a software model can be used to perform a design to determine the required combination(s) of f_{R1} and f_{R4} values, instead of fibre dosage(s).

The design process in this case is very similar to the one discussed in the previous section. An in depth explanation of the design process is therefore not given. However, a flowchart illustrating the process is provided in Appendix D.

Considering the design process outlined in Section 3.4.1.2, any adjustments previously made to fibre dosage values will in this case be made to the f_{R1} and f_{R4} values. If the values are to be increased, two increment values are calculated as shown by Equations 13 and 14 and added to f_{R1} and f_{R4} respectively.

$$increment_{5,1} = f_{R1} \times \left(\frac{M - C}{C}\right) \div 100 \quad \text{Eq. (13)}$$

$$increment_{5,2} = f_{R4} \times \left(\frac{M - C}{C}\right) \div 100 \quad \text{Eq. (14)}$$

Alternatively, the values can be decreased by the increment values suggested by Equations 15 and 16.

$$increment_{6,1} = f_{R1} \times \left(\frac{C - M}{C}\right) \div 100 \quad \text{Eq. (15)}$$

$$increment_{6,2} = f_{R4} \times \left(\frac{C - M}{C}\right) \div 100 \quad \text{Eq. (16)}$$

The design also relies on the same conceptual considerations discussed previously, although no bracket of feasible values is accounted for. For more information on this design method, see Section 3.4.1.2, as well as Appendix D.

3.5 Optimisation Algorithm

The aim of this layer is to determine a slab design which is not only sufficient to carry all loads, but which is also the most cost-effective solution possible. The process of determining this optimal solution is referred to as "optimised slab design".

The Optimisation Algorithm will regularly utilise the Design Algorithm to identify pairs of bay thickness and fibre-content values which yield sufficient capacity to carry all relevant loads.

3.5.1 Objective function derivation

An essential first step in any problem which involves optimisation is establishing a suitable objective function. As the name suggests, this function encapsulates the objectives of the optimisation process for the problem at hand. The optimisation process then endeavours to find a combination of input variables which gives a desirable - typically a minimum - result when put into the objective function.

Considering the purpose of the algorithm developed in this section, it is clear that the objective function for cost minimisation should be a function to calculate the cost of a slab. The algorithm can then be used to find values for the function variables which result in the lowest possible cost.

The cost of constructing a concrete SOG depends on several variables, including various material volumes, the cost of relevant concrete constituents such as cement, sand and plasticiser, the cost of fibre reinforcing to be used, the cost of labour and equipment to construct the slab, soil characteristics and cost of compaction. Although these variables should all be accounted for when calculating the total cost of a slab, certain practical limitations enable us to make various simplifications when deriving a cost function.

Firstly, it is assumed that the fibre product to be used in constructing a certain slab is known beforehand. Therefore, the unit cost of fibres, C_f, can be considered constant throughout the optimisation process. Similarly, the surface area of a bay in plan, A, is also seen as a constant throughout the process of slab-design. This is because the surface area is considered to be independent of the design process, and rather dependent on more practical considerations established prior to slab design.

Secondly, since the concrete material of a bay will be established prior to optimisation, the cost of concrete, C_c, is also seen as a constant when deriving the objective function. The concrete used will either be a unique material specified by the user, or a standardised material connected to a given strength class, with known characteristic values. Since the number of standardised strength classes which are typically used is very limited, it is unnecessary to include concrete type in the objective function. Instead, the slab can be optimised using each of a specified range of materials, to enable subsequent selection of the most cost effective option.

Lastly, considering the scope of this study and the objectives of the developed algorithms, the cost of construction labour and equipment is not considered by the algorithm set out in this section. For the same reason, soil characteristics and compaction are also assumed to be known and fixed prior to the start of design. The aim of the Optimisation Algorithm developed is therefore only to minimise the cost of the slab by reducing material expenditures.

Taking account of the above simplifications, an objective function is derived where all variables, excluding bay thickness, *h,* and fibre dosage, *fD*, are assigned constant values prior to each optimisation. Therefore, the function by which the cost of a slab is calculated is as follows:

$$C_T = \sum_{bays} A \cdot \frac{h}{1000} \cdot \left(C_c + fD \cdot C_f\right) \qquad \text{Eq. (17)}$$

where:

C_T = total cost of the slab

A = surface area of the current bay (constant) [m^2]

h = thickness of the current bay (variable) [mm]

C_c = unit cost of the concrete in use, per cubic meter, entered by the user (constant)

fD = fibre dosage of the current bay (variable) [kg/m^3]

C_f = unit cost of the fibres in use, per kilogram, entered by the user (constant)

3.5.2 Procedure for optimised SOG design

Considering the optimisation problem at hand, it has been found that several existing optimisation methods, including the well-known Genetic-algorithm and Particle-swarm methods are well suited to finding an optimised solution to the problem. However, since the objective function only contains two variables which can be optimised: *h* and *fD*, and since these variables are both within relatively small ranges of feasibility, it was deemed unnecessary to use a formal optimisation technique. Instead, to find the optimal solution for each bay, a customised optimisation method has been developed. This method involves iterating through a feasible range of fibre-dosages for each bay and employing the design procedure outlined in Section 3.4.1.1, to determine the required bay thickness at each iteration. The objective being to determine the most cost effective pair of fibre-dosage and bay thickness values.

A step-wise discussion of the newly developed technique is given below, and a visual illustration of the process is given in Appendix E, by means of a flowchart.

1. Establish a set of viable concrete materials, with specific unit costs, which can be used to construct the slab. Repeat steps 2 to 7 for each material.
2. Determine the feasible range within which the fibre-dosage of the bay can lay. Thus, determine an upper and lower limit for fibre-dosage. This is based either on values entered by the user, or on values calculated as discussed in Section 3.6.4, depending on which values govern.
3. Set the fibre-dosage of the bay to the lower limit value, determined in (2.).
4. Calculate the required bay thickness, based on the current fibre-dosage, to produce sufficient load capacity. This is done by using the Design Algorithm. Ensure that the calculated thickness exceeds the minimum value required by the user and/or TR34.
5. Calculate and record the material cost of the bay, based on the current combination of fibre-dosage and bay thickness.
6. Increase the fibre-dosage by an increment of 0.01 kg/m^3.
7. Repeat steps (4.) to (6.) until the fibre-dosage of the bay equals the upper limit value, determined in (2.).
8. Set the bay thickness, fibre-dosage and material to the value combination which yielded the lowest total cost, of the values tried.

3.6 Lower level algorithms

3.6.1 Determination of bay neutral-axis and moment capacity

As discussed in Section 2.3, the analysis of fibre reinforced concrete, particularly in tension, is more complex than that of normal concrete, due to its strain softening post-cracking behaviour. For this reason, accurate calculation of the moment capacity of a FRC section, as well as the location of its neutral axis, requires a rigorous assessment which takes account of the forces and fibre reinforcing across the section.

A conservative approximation of the bending-moment capacity of a fibre-reinforced SOG, M_u, can be made using (Concrete Society 2013):

$$M_u = \frac{h^2}{\gamma_m}(0.29\sigma_{r4} + 0.16\sigma_{r1}) \qquad \text{Eq. (18)}$$

where:

h = slab thickness [mm]

γ_m = material safety factor (see Section 3.6.6)

σ_{r1} = the mean axial tensile strength at CMOD 0.5 mm

σ_{r4} = the mean axial tensile strength at CMOD 3.5 mm

This relies on the assumptions discussed in the previous section.

Using the equations given in Section 2.3.4 (Equations 5 and 6), along with Equations 7 and 8, Equation 18 can be modified so that M_u is linearly equated to fibre dosage, fD, as follows:

$$M_u = \frac{h^2}{\gamma_m}(0.0589 \cdot fD + 0.0238) \qquad \text{Eq. (19)}$$

Although Equation 19 is very convenient to use, a more accurate model for calculating moment capacity exists. Appendix C of TR34 (Concrete Society 2013) provides an in-depth assessment of the forces across a FRC section and outlines the process of determining the design values, as follows.

Firstly, the position of the neutral axis of the section must be established. This is an iterative process, by which the crack height, h_c, is determined. The process works by initially setting the value of h_c to zero and then increasing it by 0.01 mm increments, until horizontal force equilibrium across the section is satisfied, with an acceptably small error. Horizontal equilibrium over a section of unit width is represented by:

$$N_1 + N_2 = T_1 + T_2 + T_3 \qquad \text{Eq. (20)}$$

where:

N_1 = Compressive force in the part of the section which is subjected to the design compressive strength of the concrete (f_{cd}).

N_2 = Compressive force in the part of the section which is not subjected to f_{cd}.

T_1 = Tension force provided by the fibres on the section – component 1.

T_2 = Tension force provided by the fibres on the section – component 2.

T_3 = Tension force provided by the steel reinforcing on the section (zero in this case).

Figure 3.3 shows the locations of the forces involved in Equation 20.

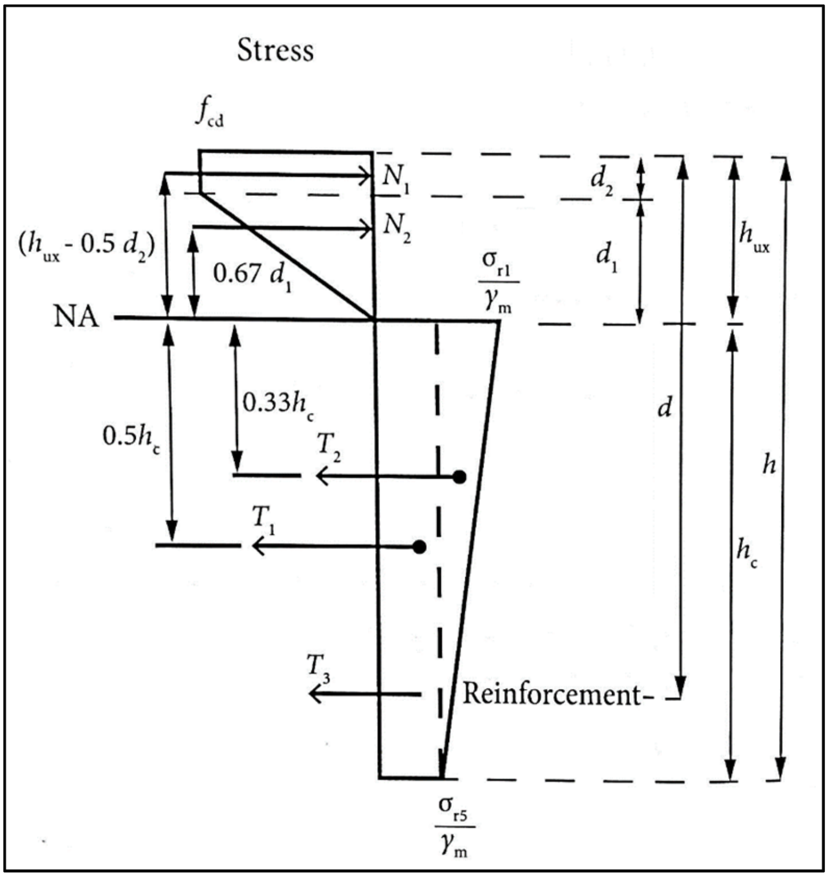


Figure 3.3: Stress diagram for a FRC section at ULS (Concrete Society 2013)

Once the value for h_c has been found, and assuming all other variables are known, the bending moment capacity of the section can be calculated using:

$$M_u = N_1 0.67 d_1 + N_2(h_{ux} - 0.5d_2) + 0.5h_c T_1 + 0.33h_c T_2 + T_3(d - h_{ux}) \qquad \text{Eq. (21)}$$

Formulae for the variables in Equation 21 are given in Appendix F.

Even though Equation 21 can be used to calculate a theoretical moment capacity for any fibre-dosage value, investigation of the predicted moment capacities has shown that it becomes less conservative than the approximated approach (Equation 19) for higher fibre dosage values. This is illustrated by Figure 3.4, which compares the moment capacity predictions of the two models, at varying fibre dosages, for a bay with thickness: $h = 300$ mm and concrete tensile strength: $f_{ctm} = 3.2$ MPa.

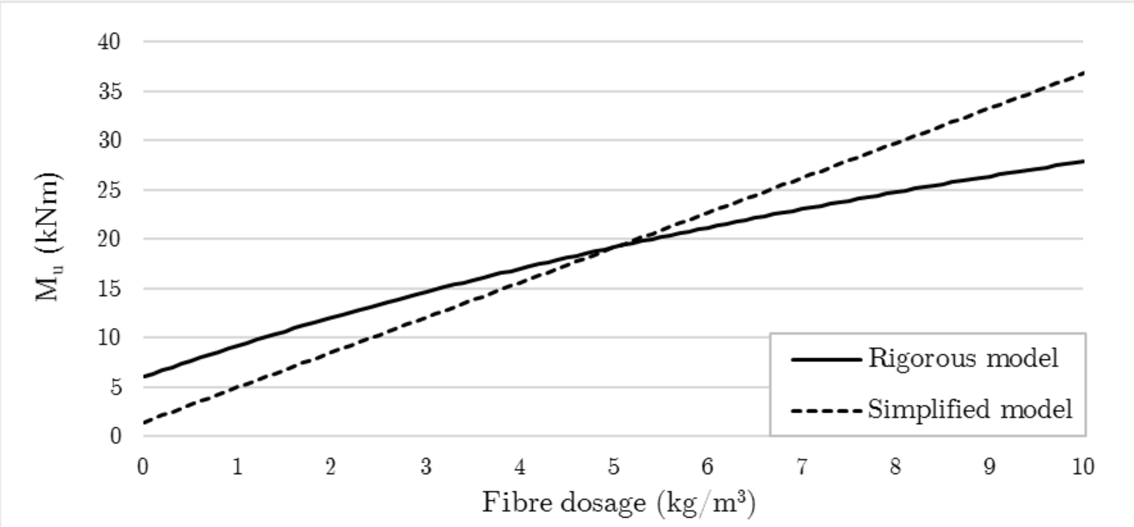


Figure 3.4: Comparison of moment capacity calculation methods

It has also been found that the increase in moment capacity, predicted by the rigorous approach, gradually stabilises as fibre dosage is increased. The practical implication of this stabilisation is that the theoretical moment capacity which the bay can reach for any fibre dosage is limited. Thus, it is possible that the capacity required for a certain bay will not necessarily be achieved by increasing the fibre dosage of that bay by any amount.

Considering the slab design methods outlined by Sections 3.4.1.1 and 3.5.2, this poses a conceptual problem, seeing as the methods rely on attaining a sufficient bay capacity by appropriately increasing fibre dosage, when required. For this reason, a unified model is implemented, which is a combination of the rigorous and simplified methods. This model works by calculating the predicted moment capacity for both methods, for a specific fibre-dosage, and selecting the larger value. This is illustrated by Figure 3.5, which shows the unification of the two curves in Figure 3.4 into a single model, which is appropriately conservative and applicable for all fibre dosages.

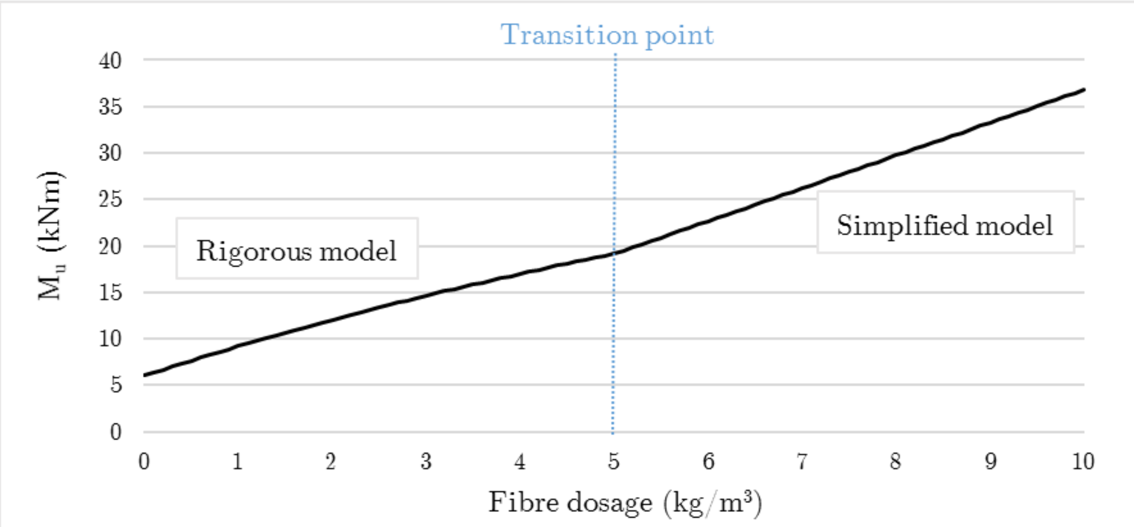


Figure 3.5: Unified moment capacity calculation model

3.6.2 Bay load capacity calculation – considering bending

The load capacity of a bay, considering a specific point-, line- and uniformly distributed loads, is calculated following the design guidelines set out in TR34 (Concrete Society 2013). These design guidelines involve the elastic and yield-line theories discussed in Section 2.4, and provide the following complete set of formulae given in this section, which predict the load capacities of FRC slabs, considering bending moments.

All load capacity formulae in this section consider the moment capacity of a fibre-reinforced bay: M_u (see Section 3.6.1) and/or the moment capacity of a plain concrete bay: M_{un}, given by:

$$M_{un} = f_{ctd,fl} \cdot \frac{h^2}{6} \qquad \text{Eq. (22)}$$

where:

h = bay thickness [mm]

$f_{ctd,fl}$ = design flexural tensile strength of concrete [MPa]

All single-, dual- and quadruple point load capacity formulae are based on yield-line theory, as given by Meyerhof (1962).

Considering point loads, unique formulae are given depending on the bay region where a load is located (see Figure 2.5) and the a/l ratio of the point load and bay under consideration, where:

a = equivalent radius of contact area of the load - see Equation 1 [mm]

l = radius of relative stiffness of the bay - see Equation 3 [mm]

<u>Single-point-loads</u>

For internal-single-point-loads:

If $a/l = 0$: $$P_{u,0} = 2\pi(M_u + M_{un}) \qquad \text{Eq. (23)}$$

If $a/l \geq 0.2$: $$P_{u,0.2} = \frac{4\pi(M_u + M_{un})}{1 - \frac{a}{3l}} \qquad \text{Eq. (24)}$$

For edge-single-point-loads:

If $a/l = 0$: $$P_{u,0} = \frac{\pi}{2}(M_u + M_{un}) + 2M_{un} \qquad \text{Eq. (25)}$$

If $a/l \geq 0.2$: $$P_{u,0.2} = \frac{\pi(M_u + M_{un}) + 4M_{un}}{1 - \frac{2a}{3l}} \qquad \text{Eq. (26)}$$

For corner-single-point-loads:

If $a/l = 0$: $$P_{u,0} = 2M_{un} \qquad \text{Eq. (27)}$$

If $a/l \geq 0.2$: $$P_{u,0.2} = \frac{4M_{un}}{1 - \frac{a}{l}} \qquad \text{Eq. (28)}$$

For a/l values between 0 and 0.2, linear interpolation is used to determine relevant point load capacities.

<u>Dual-point-loads</u>

The theoretical dual-point-load capacity of a bay also relies on yield line theory (Meyerhof 1962) and takes account of the distance, x, between the two single-point-loads.

The dual-point-load capacity shall never exceed the sum of the two individual single-point-load capacities.

An internal dual-point-load is one where both single-point-loads are located within the internal region of the bay. If either of the loads is located within the corner region, the dual-point-load is considered as a corner load. Otherwise, if either of the loads is located within the edge region, the dual-point-load is considered as an edge load.

For internal-dual-point-loads:

If $a/l = 0$:
$$P_{u,0} = \left[2\pi + \frac{1.8x}{l}\right] \cdot [M_u + M_{un}] \qquad \text{Eq. (29)}$$

If $a/l \geq 0.2$:
$$P_{u,0.2} = \frac{4\pi(M_u + M_{un})}{1 - \frac{a}{3l}} \qquad \text{Eq. (30)}$$

For edge- and corner-dual-point-loads:

Since no equations exist for dual-point-loads outside the internal region of a bay, a simplified approach is taken. The edge-dual-point-load capacity is calculated by computing the internal capacity, using Equation 29 and/or 30, and reducing the result by the ratio of internal to edge capacities for single-point-loads. The corner-dual-point-load capacity is calculated through the same process; however, using the ratio of internal to corner capacities for single-point-loads.

Quadruple-point-loads

Load capacity calculations for quadruple-point-loads rely on the same assumptions as those for dual-point-loads. However, it is necessary to account for the distance between the constituent single-point-loads in two directions: x and y, yielding the following equations:

For internal-quadruple-point-loads:

If $a/l = 0$:
$$P_{u,0} = \left[2\pi + \frac{1.8(x+y)}{l}\right] \cdot [M_u + M_{un}] \qquad \text{Eq. (31)}$$

If $a/l \geq 0.2$: $$P_{u,0.2} = \left[\frac{4\pi}{1 - \frac{a}{3l}} + \frac{1.8(x+y)}{l - \frac{a}{2}}\right](M_u + M_{un})$$ Eq. (32)

For edge- and corner-quadruple -point-loads:

The same simplified approach taken for edge- and corner-dual-point-loads applies to quadruple-point-loads.

<u>Line loads</u>

The line load capacity formulae given below are based on elastic theory, as given by Hetényi (1971). These formulae involve the term: λ (see Equation 2), as well as the moment resistance of plain, unreinforced concrete: M_{un} (see Equation 22). Due consideration of the bay regions affected by a line load is also essential. If a line load stretches across multiple regions, the load is divided into segments, as described in Section 3.6.5. Each segment is then analysed individually.

For internal line load segments:

$$P_{lin} = 4 \cdot \lambda \cdot M_{un}$$ Eq. (33)

For edge line load segments:

$$P_{lin} = 3 \cdot \lambda \cdot M_{un}$$ Eq. (34)

For middle line load segments:

Line load capacities within the middle region of the bay are calculated using linear interpolation between the values calculated using Equations 33 and 34. This interpolation is done by considering the point on the line load, or line load segment, which is closest to the edge region of the bay.

<u>Uniform distributed loads</u>

Similar to line load capacity calculations, the following UDL capacity formula is derived from elasticity theory (Hetényi 1971) and takes the terms λ and M_{un} into account. However, UDL capacity is considered constant at all points on a given bay; therefore, knowledge of the location of the load is unnecessary.

$$q = 5.95 \cdot \lambda^2 \cdot M_{un}$$ Eq. (35)

3.6.3 Bay load capacity calculation – considering shear

Calculation of the load capacity of a bay depends on the ability of the bay to resist bending- and shear-failure respectively. When considering point loads, punching shear capacity is determined in line with Eurocode 2 (British standards institution 2004a), by considering the shear stress at the face of the loaded area (Equation 36), as well as at a critical perimeter, 1.5h from the face of the contact area (Equation 37); where h is the bay thickness. The part/percentage of the reaction force carried by the soil directly below the load is taken into account at a later stage.

$$P_{p,max} = v_{max} \cdot A_f \qquad \text{Eq. (36)}$$

$$P_p = (v_{Rd,c} + v_{fib}) \cdot A_c \qquad \text{Eq. (37)}$$

where:

$P_{p,max}$ = maximum load capacity in punching, considering the face of the loaded area [N]

P_p = maximum load capacity in punching, considering the critical perimeter [N]

v_{max} = maximum allowable shear stress [MPa]

$v_{Rd,c}$ = minimum shear strength of unreinforced concrete [MPa]

v_{fib} = increase in shear strength provided by reinforcement [MPa]

A_f = shear face area, at the edge of the load [mm^2]

A_c = shear face area, at the critical perimeter [mm^2]

While the values of v_{max} and $v_{Rd,c}$ depend on material properties and are therefore constant throughout a given bay, calculation of the shear face areas: A_f and A_c, requires some geometric consideration for each individual point load. These areas are calculated as the product of the effective depth of the bay: $d = 0.75h$, and the length of the perimeter under consideration: u_f or u_c. It should be noted that, since macro-synthetic fibres do not provide reinforcement against shear failure, v_{fib} is taken as zero throughout this project.

Considering the generalised method(s) used to calculate load capacities during slab analysis (see Section 3.3.1), standardised methods of determining u_f and u_c for each point-load type is also warranted. Figures 3.6 to 3.8 illustrate the load perimeters of different types of point-loads within the inner-, edge- and corner-regions of a bay, along with the equations used to calculate the perimeter dimensions. These are conservative values, which depict the "worst-case" scenario, in terms of shearing action, for point-loads located in the various bay regions.

Column base point loads

Since column base plates are typically rectangular, their perimeter measurements within the internal region of a bay are calculated as the sum of the straight lines around their edges: w and b, as shown in Figure 3.6. However, the fact that these rectangular plates are often not oriented flush to bay edges means that a simplification is needed when calculating the perimeter measurements within the edge and corner regions. For this reason, the perimeters of a column base within the edge or corner zones are calculated using the centre point of the load, along with its equivalent radius: a (see Equation 1) and modified radius: r, where:

$$r = a + 1.5h \qquad \text{Eq. (38)}$$

Truck wheel point loads

The loads from vehicle wheels on the slab are typically taken as circular. Thus, all wheel load perimeters are calculated using the centre point of the load and unique a and r values, as shown in Figure 3.7.

Combined point loads

The approximated shape of a combined point load, consisting of two pre-existing single point loads, is shown in Figure 2.4. The perimeter of this approximated shape, P_f, can be calculated for any combined point load, using its a_1 and a_2 values, along with the distance between the loads' centre points. In a similar manner, a critical perimeter, $1.5h$ from P_f, can be calculated as P_c. Therefore, load perimeters for combined loads are calculated by considering P_f or P_c, and the maximum value of a_1 and a_2: a_{max}, as shown in Figure 3.8. The perimeter values given in Figure 3.8 shall not exceed the sum of the perimeters for the individual loads, for their respective bay regions.

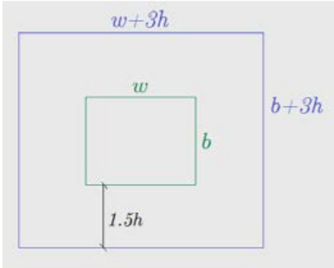
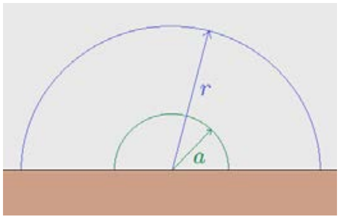
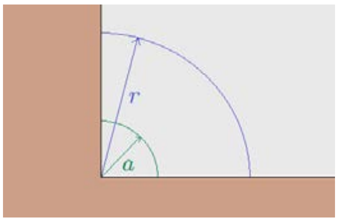
Bay region	**Load edge perimeter (u_f)**	**Critical perimeter (u_c)**	**Illustration**
Internal	$2(w+b)$	$2(w+b+6h)$	
Edge	$\pi \cdot a$	$\pi \cdot r$	
Corner	$0.5\pi \cdot a$	$0.5\pi \cdot r$	

Figure 3.6: Shear perimeters for column base point loads

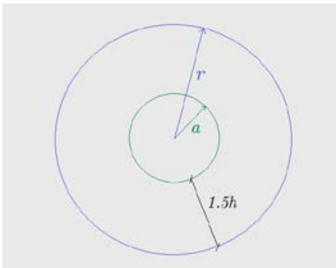
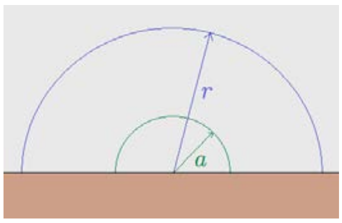
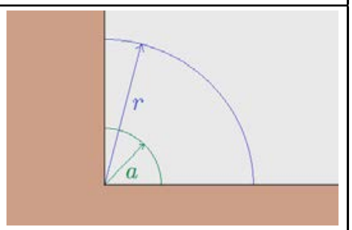
Bay region	**Load edge perimeter (u_f)**	**Critical perimeter (u_c)**	**Illustration**
Internal	$2\pi \cdot a$	$2\pi \cdot r$	
Edge	$\pi \cdot a$	$\pi \cdot r$	
Corner	$0.5\pi \cdot a$	$0.5\pi \cdot r$	

Figure 3.7: Shear perimeters for truck wheel point loads

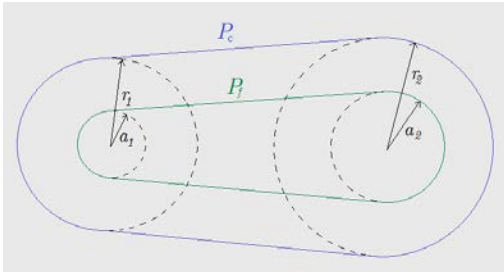
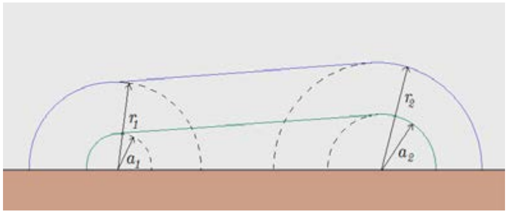
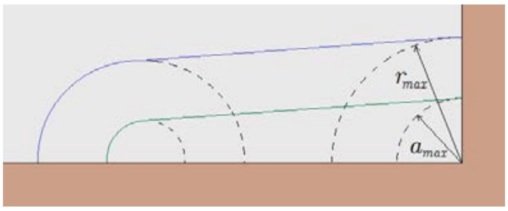
Bay region	**Load edge perimeter (u_f)**	**Critical perimeter (u_c)**	**Illustration**
Internal	P_f	P_c	
Edge	$0.5P_f$	$0.5P_c$	
Corner	$0.5P_f$ - $0.5\pi \cdot a_{max}$	$0.5P_c$ - $0.5\pi \cdot r_{max}$	

Figure 3.8: Shear perimeters for combined point loads

Line- and uniform distributed loads

It is assumed that shear action resulting from line- and uniform distributed loads is negligible. This is done through consideration of elasticity theory, and the fact that these loads are not highly concentrated, by definition.

3.6.4 Feasible fibre dosage values for slabs-on-grade

In the design of slabs-on-grade, the amount of fibres being used will either be provided by the user or calculated based on various other factors. However, for both these cases, it is not guaranteed that the fibre dosage will be within reasonable bounds, from a construction point of view. Thus, it is necessary to specify a range of values for which fibre dosage is acceptable, so that all values outside the range can be adjusted to ensure constructability.

Minimum fibre dosage

The Concrete Society (2004) advises that the fibre content of ground-supported slabs should be such that the ratio of cracked to un-cracked moments of resistance always exceeds 50%. Thus, taking these values to be the moment resistances of a reinforced and unreinforced concrete section respectively, we obtain the following equation:

$$\frac{M_u}{M_{un}} \geq 0.5 \qquad \text{Eq. (39)}$$

Substituting the formulae for M_u and M_{un} respectively (given by TR34) into the above equation yields:

$$\frac{\frac{h^2}{\gamma_m}(0.29\sigma_{r4} + 0.16\sigma_{r1})}{\frac{f_{ctm}}{\gamma_m}\left(1.6 - \frac{h}{1000}\right) \cdot \frac{h^2}{6}} \geq 0.5 \qquad \text{Eq. (40)}$$

Next, substituting $[\sigma_{r1} = 0.45 f_{R1}]$ and $[\sigma_{r4} = 0.37 f_{R4}]$ and then simplifying, we obtain:

$$\frac{0.1073 f_{R4} + 0.072 f_{R1}}{\frac{f_{ctm}}{6}\left(1.6 - \frac{h}{1000}\right)} \geq 0.5 \qquad \text{Eq. (41)}$$

Lastly, using the equations given by Bester (2017) (see Section 2.3.7) we obtain an expression for minimum fibre dosage, fD_{min}, based on slab thickness, h, and mean axial tensile strength of concrete, f_{ctm}:

$$fD_{min} = 1.4136 \cdot f_{ctm}\left(1.6 - \frac{h}{1000}\right) - 0.4036 \qquad \text{Eq. (42)}$$

For the sake of convenience, we rewrite Equation 42 in terms of the characteristic cylinder compressive strength of the concrete (f_{ck}), using $\left[f_{ctm} = 0.3 \cdot {f_{ck}}^{2/3}\right]$. Thus:

$$fD_{min} = 0.4241 \cdot {f_{ck}}^{2/3}\left(1.6 - \frac{h}{1000}\right) - 0.4036 \qquad \text{Eq. (43)}$$

In order to consider the practical implications of Equation 43, Table 3.1 shows the fibre dosage values, fD_{min}, required for a range of typical slab thickness, h, and concrete strength, f_{ck}, combinations. It shows that the amount of fibres needed to ensure adequate ductility in a ground-supported slab decreases as the thickness of the slab increases, and increases for larger values of concrete strength.

Maximum fibre dosage

Apart from high material costs, there are other factors which can limit the amount of fibres in a FRC slab-on-ground. These include balling of the fibres and possible reduction of concrete workability.

Different from the minimum fibre dosage, the maximum fibre dosage value (fD_{max}) for a SOG will typically not be based on theoretical or numerical computations, but rather on experimental investigation or knowledge of practise. Therefore, the fibre supplier will normally provide a recommended maximum fibre dosage, for a specific fibre product.

Table 3.1: Fibre dosages required for typical slab thickness and concrete strength values.

$fD_{\min}$ [kg/m^3]		f_{ck} [MPa]					
		25	28	30	32	35	40
Thickness (h) [mm]	150	4.85	5.27	5.53	5.79	6.18	6.79
	160	4.82	5.23	5.49	5.75	6.13	6.74
	170	4.78	5.19	5.45	5.71	6.09	6.69
	180	4.75	5.15	5.41	5.67	6.04	6.64
	190	4.71	5.11	5.37	5.62	5.99	6.59
	200	4.67	5.07	5.33	5.58	5.95	6.54
	210	4.64	5.03	5.29	5.54	5.90	6.49
	220	4.60	4.99	5.25	5.50	5.86	6.44
	230	4.56	4.95	5.21	5.45	5.81	6.39
	240	4.53	4.91	5.17	5.41	5.77	6.34
	250	4.49	4.88	5.12	5.37	5.72	6.29
	260	4.46	4.84	5.08	5.32	5.68	6.24
	270	4.42	4.80	5.04	5.28	5.63	6.19
	280	4.38	4.76	5.00	5.24	5.59	6.14
	290	4.35	4.72	4.96	5.20	5.54	6.09
	300	4.31	4.68	4.92	5.15	5.50	6.04
	310	4.27	4.64	4.88	5.11	5.45	6.00
	320	4.24	4.60	4.84	5.07	5.40	5.95
	330	4.20	4.56	4.80	5.03	5.36	5.90
	340	4.17	4.52	4.76	4.98	5.31	5.85
	350	4.13	4.48	4.71	4.94	5.27	5.80
	360	4.09	4.45	4.67	4.90	5.22	5.75
	370	4.06	4.41	4.63	4.85	5.18	5.70
	380	4.02	4.37	4.59	4.81	5.13	5.65
	390	3.98	4.33	4.55	4.77	5.09	5.60
	400	3.95	4.29	4.51	4.73	5.04	5.55

3.6.5 Division of line loads into segments

The fact that all bays comprise of various regions, with varying load capacities, is discussed in Section 2.2.2. To ensure accurate results, it is important to consider the region of a bay where any line load acts, when calculating the capacity of the bay to carry the specific load.

Since line loads often act across multiple bays and regions, a method of dividing such loads into appropriate segments is outlined in this section. The capacity of the slab to carry each line load segment can then be calculated. Such segments traverse the interior, middle or edge zone(s) of the bay(s) which are affected by the load. Each segment lies within a single region of a particular bay, as illustrated by Figure 3.9.

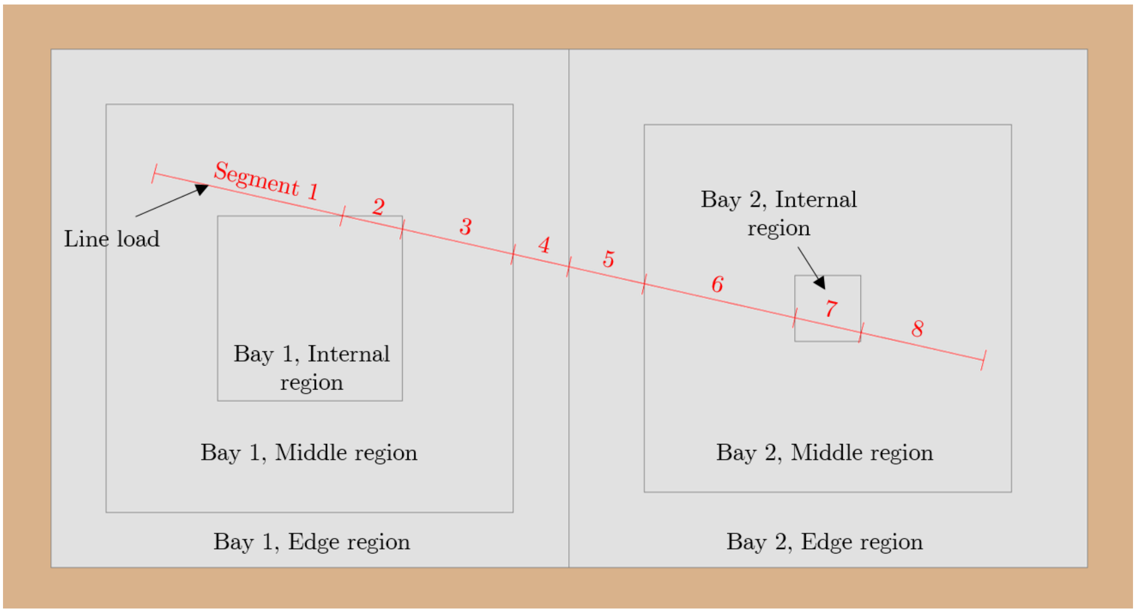


Figure 3.9: Segments of a line load which acts across various regions of multiple bays

The process by which a line load is divided into segments is as follows:

1. Identify all bays which are affected by the line load.
2. Establish the perimeters of the relevant regions which make up the affected bays.
3. Assign a list of critical points to the line load and record the line load intensity at each critical point.

3.1 The first and last critical points are the start and end points, respectively, of the line load.

3.2 The intermediate points are where the line load intersects any of the perimeters defined in (2.).

4. Starting at the first critical point, establish a line load segment between each pair of sequential critical points. Record the start and end points, the start and end intensities and bay region of each line load segment.

3.6.6 Partial safety factors

TR34 stipulates the partial safety factors, γ, for ground supported slabs as follows:

Materials (γ_m):

Concrete	-	1.5	
Concrete with fibre	-	1.5	
Bar or fabric reinforcement	-	1.15	

Loads (γ_F):

Defined racking	-	1.2	
Other	-	1.5	
Dynamic loads	-	1.6	
Line loads and UDLs	-	1.5	(1 if a material SF is already used)

If a mezzanine is present and supported on the slab, a partial safety factor of 1.35 should be used for the dead load of the mezzanine structure and 1.5 for any imposed loads on the mezzanine.

The above safety factors are generally less conservative than those set out by the ACI (2010).

Chapter 4

Proposed software model

Using the various conditional calculations, checks and procedures outlined in TR34 and its associated documentation, a three-part slab-design algorithm has been compiled which takes comprehensive account of all possible variables (see Chapter 3). This chapter serves to provide an overview of the proposed software model, which is capable of implementing the developed set of algorithms by computerised means. Constraints and procedures which the model must adhere to and provide for are also identified and certain unique concepts are described in detail.

The process of initialising, analysing and designing a FRC slab-on-ground, by means of a software prototype, follows the guidelines set out in Sections 4.2 to 4.5, depending on the objectives of the user.

4.1 Design objectives

Following the aims of TR34 (2013), the primary design objectives are for the slab to carry all intended loads and to avoid surface cracking. The latter objective, although being quite simple in concept, warrants special consideration in light of the yield-line theory basis of design.

Considering the majority of loads which will typically be placed on a SOG, the ultimate strength modes of failure are identified as flexure and punching. While design against punching shear is based on relatively simple conceptions, the vital objective of avoiding surface cracking creates some complications when considering flexure.

Yield-line consideration of a slab at ULS assumes plastic behaviour, aided by sufficient ductility. Thus, the bending moment along the positive/sagging yield lines is assumed to be the residual post-cracking value. However, seeing as upper surface cracking of the slab is to be avoided, this same assumption cannot be made for the negative/hogging yield lines. Therefore, the bending moment of the slab along the negative yield lines is limited by basing it on the design cracking strength of the concrete, neglecting the reinforcing strength of the fibres, with an appropriate safety factor. Since the slab will not actually fail when the design negative moment is reached, the design does not truly consider the ULS. To enable this simplification, two additional assumptions are required (Concrete Society 2013):

Firstly, at the apparent ULS, it is assumed that the limiting compressive strength of the concrete, either at the bottom or at the top of the slab, is reached simultaneously with the limiting tensile strength of the fibre concrete, at the opposite side of the concrete. This guarantees strain compatibility, but not force equilibrium.

Secondly, neutral axis location is considered to be directly related to slab depth.

In a typical slab-design situation, the above assumptions allow the designer to make conservative approximations of the ultimate moment capacity of the slab. This is very convenient, as the number of calculations needed are significantly reduced. However, once a software implementation of the problem is established, these simplifying assumptions are, in some cases, unnecessary and overly conservative. Thus, calculation of the bay neutral axis location and bending moment capacity is done in a more rigorous fashion, as discussed in Section 3.6.1.

4.2 Initialisation of slab attributes and bays

Preceding the analysis and design of a slab, initialisation of all physical properties and loading information must be completed. The sequence in which objects and loads are added to the model is not strictly prescribed; however, certain coded restrictions guide the user to input data in an order that is sensible. For example, no loads can be added to the slab before at least one bay has been created. Chapter 6 contains more information on the user interface of the program and its operation.

When the software prototype is started, the user is prompted to input several of these parameters. However, most of them can instead be entered and/or modified at a later stage.

The properties, which should be set before an analysis or design can be performed include:

- An identifier for the slab
- Universal slab properties. These are bay properties that will be constant across the entire slab, i.e. for all bays. These are optional attributes, meaning that any number of them can be set or omitted.
  - Thickness: h
  - Modulus of subgrade reaction: k
  - Material/concrete
  - Fibre specifications: either fibre type and dosage, or f_{R1} and f_{R4} values
- At least one bay object must be created, and each bay should have all of the following characteristics defined. If any attributes have been set as universal (see above), they are automatically assigned to bays during creation.
  - Bay name
  - Thickness: h
  - Modulus of subgrade reaction: k
  - Material/concrete
  - Fibre specifications: either fibre type and dosage, or f_{R1} and f_{R4} values

  For rectangular bays:

  - A reference point: coordinates of the lower left-hand corner
  - Width and length of the bay

  For polygonal bays:

  - A set of points, specified by coordinates, defining the outline of the bay

  When a bay is added to the slab, the boundaries of its corner- edge- and internal-regions; considering point-loads, as well as its edge- and middle-regions; considering line-loads are automatically determined.

Once the above information has been set, analysis and design of the slab can be carried out. However, this will not yield any useful results if the relevant loads have not yet been defined.

4.3 Initialisation of loads

As mentioned in Section 2.2.2, the main load types under consideration are point, line and uniformly distributed loads. Any number and combination of these loads can be added to the slab, as objects, in the following manner:

Point loads:

Two types of single point load can be added to the slab, and requires input of the data stated:

- Column base point load(s)
  - Load name
  - Coordinates of load centre
  - Load magnitude
  - Amount and spacing, in X and Y directions, of duplicate loads to be added
  - Column base plate dimensions: width, length and thickness
  - Column cross-sectional dimensions: either width and height, or diameter
- Truck wheel point load(s)
  - Load name
  - Coordinates of load centre
  - Truck/vehicle mass
  - Tyre pressure, only for pneumatic tyres
  - Contact surface area, only for solid tyres

  Truck wheel point loads can also be added to the slab by means of a traffic zone, as described later on.

After the single point loads have been added, the software prototype can be used to convert sets of loads into combined, dual and/or quadruple loads, if they are within certain distances of each other.

Line loads:

To create a new line-load object, the following data should be entered:

- Line load name
- Coordinates of the start and end points
- Load intensity at the start and end points. Linear load variation between the two points is assumed.

Uniform distributed loads:

To create a new UDL object, the following data should be entered:

- UDL name
- Load magnitude

For rectangular UDLs:

- A reference point: coordinates of the lower left-hand corner
- Width and length of the UDL

For polygonal UDLs:

- A set of points, specified by coordinates, defining the outline of the UDL

Traffic zones:

A traffic zone is a region on the slab where a particular vehicle is likely to move. As deliberated in Section 4.6, traffic zone objects have the ability to automatically generate stationary wheel point loads at suitable positions within the zone. To create a new traffic zone object, the following data should be entered:

- Traffic zone name
- Truck/vehicle mass
- Tyre pressure: only for vehicles with pneumatic tyres
- Contact surface area: only for vehicles with solid tyres
- Vehicle dimensions: front and rear axle widths and front-to-rear axle length

For rectangular traffic zones:

- A reference point: coordinates of the lower left-hand corner

- Width and length of the traffic zone

For polygonal traffic zones:

- A set of points, specified by coordinates, defining the outline of the traffic zone

4.4 Slab-on-grade analysis model

The algorithm by which slabs are analysed is discussed in Section 3.3. As is mentioned, the analysis procedure is based on the guideline set out in TR34 (Concrete Society 2013).

Once all bay and load data have been set, the slab can be analysed according to a specified series of steps. Through this analysis, the magnitude of each load on the slab is compared to the capacity of the relevant bay(s) to carry the specific load type. Therefore, all excessive and/or governing loads can be identified. Alternatively, identification of bays with excessive fibre contents and/or thicknesses can also be done.

4.4.1 Program output following slab analysis

If analysis of the slab is the user's current objective, a structured slab analysis report will be generated, as demonstrated in Section 6.10, stating all relevant information pertaining to the bays and loads of the slab, including the identification of limiting and excessive loads.

For each type of load, a section is included in the report stating:

- Whether or not the capacity of the slab for the specific load type has been exceeded.
- All loads which exceed the capacity of the slab.
- The load(s) that currently govern the capacity of the slab for the specific type. Also state the bay and region where the load(s) are located.
- A list of all loads, categorised by type, stating the basic information of each load, the capacity of the slab to carry it in shear and bending, and whether or not it exceeds this capacity.

4.5 Slab-on-grade design model

The algorithms by which slabs are designed are discussed in Sections 3.4 and 3.5. Two main types of design; namely, basic and optimised design, are identified and explained.

After the relevant bay and load data have been entered, the slab can be designed to establish a set of bay characteristics which will provide load capacity which is sufficient, but not excessive. Additionally, the following information can also be specified; however, provided default values can also be used:

For all types of design:

- Minimum slab thickness. The default value is 150 mm, based on clause 7.1 of TR34.
- Minimum fibre dosage. The default value is 1 kg/m^3, based on product information for CHRYSO®Fibre S50 fibres.
- Maximum fibre dosage. The default value is 8 kg/m^3, based on the same product information.

For optimised design:

- Unit cost of concrete. An arbitrary default value of 80 €/m^3 is given.
- Unit cost of fibres. An arbitrary default value of 50 €/kg is given.

Any one of the following types of design can be performed, depending on the objective of the user. A reference to the algorithm describing each design process is given in parentheses.

- Basic SOG design
    - Basic slab design to suitable thicknesses (Section 3.4.1.1)
    - Basic slab design to suitable fibre dosages (Section 3.4.1.2)
    - Basic slab design to suitable f_{R1} and f_{R4} combinations (Section 3.4.1.3)
- Optimised SOG design (Section 3.5.2)

If SOG design is the user's objective, a similar report to that mentioned in Section 4.4 will be generated subsequent to the design, containing all design values and checks for sufficient load capacity.

4.5.1 Program output following slab design

Basic slab design

After completion of any of the basic design procedures mentioned in Section 4.5, the software prototype will open a slab analysis window (see Section 6.10) containing the following information:

- The type of slab design which has been performed.
- Considering the values not being designed for:
  - Any adjustments made to ensure that the values not being designed for are within feasible ranges, along with the new, post-adjustment, values.
  - Any adjustments made to standardise the values not being designed for, if relevant, along with the new, post-adjustment values.
- The design value(s) obtained, depending on the type of design performed.
- Considering the values obtained through the relevant design process:
  - Any adjustments made to ensure that the obtained design values are within feasible ranges, along with the new, post-adjustment values.
  - Any adjustments made to standardise the obtained design values, if relevant, along with the new, post-adjustment values.
- A complete slab-analysis report, as is outlined in Section 4.4.1.

Optimised slab design

After completion of the optimised design procedure mentioned in Section 4.5, the software prototype will open a slab analysis window (see Section 6.10) containing the following information:

- The type of slab design which has been performed.
- The design material, thickness and fibre dosage value(s) obtained.
- Considering the values obtained through the design process:
  - Any adjustments made to standardise the obtained design values, if relevant, along with the new, post-adjustment values.
- An estimate of the total material cost, in the selected currency, of the slab.
- A complete slab-analysis report, as is outlined in Section 4.4.1.

4.6 Automated traffic-zone wheel-point-load generation

After defining a traffic zone and specifying its associated vehicle information (see Section 4.3), a number of truck wheel point loads will automatically be placed on the slab within the zone. One or more of these loads will be created, using input vehicle data, and placed at positions which represent the worst-case scenario from a failure perspective, considering pre-existing loads and joints located within or near the traffic zone.

The process of determining appropriate positions for the loads of each specific traffic zone, is completed through the following series of steps. It is important to note that these steps, as well as the flowchart given in Appendix G, represent a simplified conceptual overview of the actual algorithm. Several utilities employed during this process are not discussed in depth, for practical reasons. If a more detailed insight into these procedures is required, examination of the relevant source code is recommended.

1. Separate all Combined-, Dual- and Quadruple point loads on the slab into their single point load components.
2. Identify all single-point-loads which are located within the traffic zone, or which are at a distance less than $2h$-a from the zone boundary. Wheel loads which were previously created by the current traffic zone are not considered.

   Repeat steps 3 to 6 for each appropriate single-point-load, thereby creating between one and four wheel loads corresponding to each relevant single-point-load.
3. Place Wheel Load 1. This is the wheel load closest to the current single-point-load and is considered to be a front wheel of the traffic zone vehicle.
   - 3.1. Position Wheel Load 1 directly next to the current single-point-load, through the process outlined in Appendix G.
   - 3.2. If another load already exists at the proposed location for Wheel Load 1, this new load is not added to the slab.
   - 3.3. If no acceptable location for Wheel Load 1 can be established, this new load is neglected and addition of Wheel Loads 2 to 4 is avoided.

4. Place Wheel Load 2. This is also considered to be a front wheel of the traffic zone vehicle.
    4.1. Position Wheel Load 2 at a distance equal to the front axle width of the vehicle from Wheel Load 1. This is done through a process very similar to the one outlined in Appendix G, however using Wheel Load 1 as a reference.
    4.2. If another load already exists at the proposed location for Wheel Load 2, this new load is not added to the slab.
    4.3. If no acceptable location for Wheel Load 2 can be established, this new load is neglected and addition of Wheel Loads 3 and 4 is avoided.
5. If Wheel Loads 1 and 2 have been added on opposite sides of the current single-point-load, the positions of these wheel loads must be adjusted, as this situation is impossible in real life.
    5.1. Move Wheel Loads 1 and 2 in a direction perpendicular to the front axle of the vehicle, to the side of the current single-point-load where the traffic zone perimeter is furthest away.
6. Place Wheel Loads 3 and 4. These are considered to be the rear wheels of the traffic zone vehicle.
    6.1. Position Wheel Loads 3 and 4 at a distance equal to the front-to-rear axle length of the vehicle from Wheel Loads 1 and 2. Wheel Loads 3 and 4 are placed at a distance equal to the rear axle width of the vehicle from each other. This is done through a similar process to the one outlined in Appendix G, however using Wheel Loads 1 and 2 and the current single-point-load as references.
    6.2. If another load already exists at the proposed location for Wheel Loads 3 or 4, the relevant new load is not added to the slab.

An example of the implementation of this algorithm by the software prototype is illustrated in Section 6.3.5, which shows six traffic-zone wheel point loads (Figure 6.12), generated through consideration of two column base point loads on the perimeter of the relevant traffic zone.

Chapter 5

Object model for SOG analysis and design

As discussed in Section 1.2, objectives B and C of this study are to develop a software prototype capable of analysing, designing and optimising a fibre-reinforced SOG. In order to develop source code for such a prototype, a logically structured object model is required, which maps the physical problem – defining, analysing and designing a slab – to the computer. Considering the scale and relative complexity of the program: approximately 20,000 lines of source code, it is essential that the model be logically structured in a way that can be understood and possibly expanded in the future.

This chapter provides an overview of the structure of the developed object model, as well as some more detailed information on the operation and logic of certain components. If knowledge of the exact workings of the software is required, examination of the source code is recommended. For clarification of the definitions of programming terms and acronyms, see the Nomenclature and Glossary sections at the start of this document.

5.1 Model layout and features

The object model has been developed to mirror the concepts involved in the physical SOG: slabs, bays, bases, materials, loads and traffic zones. To this end, the following classes have been developed in the Java programming language.

5.1.1 Physical object classes

These classes represent the physical components of a fibre-reinforced SOG.

- Slab
- Bay
- Material

- Base
- Fibre

As can be expected, a Slab object is meant to contain a set of one or more Bay objects. Each Bay object is assigned a Material, Base and Fibre. A Material object represents a type of concrete and a Fibre object contains all relevant information regarding the fibre-reinforcing in the bay, including either a fibre type and dosage, or f_{R1} and f_{R4} values.

Appendix H provides a unified modelling language (UML) diagram of the five classes listed above, illustrating the relationship between them, along with a summarised overview of their attributes and methods. Key functionalities provided by these classes are described in Section 5.2.

5.1.2 Load object classes

These classes represent the several types of loads which can act on a SOG. The name of each class indicates the type of load which it represents and multiple objects of each class can be instantiated. Although a traffic zone (see Section 6.3.5) does not represent a load, it is mentioned in this section as its main purpose is to generate loads. Loads which are generated by class TrafficZone are objects of class TZWheelPL, which is an extension of WheelPL.

Inheritance is gainfully employed by these classes, to simplify the way in which they are set out; specifically when considering the point-load classes. Multiple layers of Super- and Sub-classes help categorise the loads so that there is no unnecessary repetition of attributes and methods.

- PointLoading
    - SinglePointLoad
        - ColumnBasePL
        - CombinedPL
        - WheelPL
            - TZWheelPL
    - DualPointLoad

  - QuadruplePointLoad
- LineLoad
- LineLoadSegment
- UDL
- TrafficZone

Appendix I provides UML diagrams of the classes listed above, illustrating the relationship between them, along with a summarised overview of their attributes and methods. More detail about the facilitation of load application provided by these classes is given in Section 5.2.

5.1.3 Utilities

The class below is created to group together utility methods which perform geometric, mathematical and/or file-handling functions. This class has no attributes and objects of this class cannot be instantiated. The methods are static so that they can easily be accessed from any point in the software. A UML diagram of the class is given in Appendix J, which provides an overview of its methods.

- SlabUtils

5.1.4 GUI component classes

The following classes have been created to provide an effective graphical interface between the user and the program. Even though these classes contain some methods which collect input data and coordinate the analysis and design processes, they are not essential to the design model set out in Chapter 4. They do, however, support user-friendly gathering of input and interpretation of results. A run-through of the operation of this interface is given in Chapter 6.

Each of the following classes is used to generate a single GUI window, for the purpose indicated by the brackets following each class. The "Listener" design pattern of the Java language is used to implement the functionalities of the various GUIs.

- GUI_Start (see Section 6.2 - Welcome/initialisation window)

- GUI_SlabEditor (see Section 6.3 - Slab editor window)
- GUI_Material (see Section 6.4 - Material editor window)
- GUI_Fibre (see Section 6.5 - Fibre editor window)
- GUI_Bay (see Section 6.6 -Bay editor window)
- GUI_ColumnPL (see Section 6.7.1 - Column base point load editor window)
- GUI_WheelPL (see Section 6.7.2 - Truck wheel point load editor window)
- GUI_UDL (see Section 6.8 - UDL editor window)
- GUI_TZ (see Section 6.9 - Traffic zone editor window)
- GUI_SlabAnalyse (see Section 6.10 - Slab analysis window)
- GUI_SlabDesign (see Section 6.11 - Slab design window)
- GUI_GetNewName (see Section 6.12 - Slab rename window)
- GUI_Close (see Section 6.13 - Program exit window)

Appendix K provides a UML diagram of the classes listed above, illustrating the relationship between them, along with a summarised overview of their attributes and methods. More detail about the gathering of input data and control of the analysis and design procedures through the use of the various GUIs is provided in Section 5.2.

5.2 Key classes, attributes and methods

The analysis, design and optimisation of a SOG is described in Chapter 3. Using the procedures described there as a framework, some of the key classes, attributes and methods involved in the process are identified and discussed in this section.

5.2.1 Initialisation of slab attributes, bays and loads

Collection of all input data from a user is done by the various GUI component classes listed in Section 5.1.4. These GUI classes regularly interact with the physical and load object classes throughout the operation of the software prototype.

This section describes the classes and methods involved in initialising the physical slab components and loads. Removal of any of these objects from the slab can be done using class GUI_SlabEditor, along with the various remove(Object) methods of class Slab.

Slabs

Instantiation of a Slab object takes place within the *actionPerformed* method of class GUI_Start. Slabs can be instantiated in one of three ways: without any attributes, with only a given name, or with a name and one or more standard attributes, depending on the objectives of the user. After instantiation of the Slab, class GUI_SlabEditor can be used to add, edit or remove standard slab attributes, using the various set- methods of class Slab. It can also be used to initiate any of the following functions.

Bays

To add a new bay to the slab, a blank Bay object is instantiated and sent to the class GUI_Bay. This class then collects all relevant data and assigns it to the bay using applicable set- methods, given that no errors are encountered. It then adds the completed bay object to the slab by means of the add(Bay) method of class Slab. If an existing bay is to be edited, its complete Bay object is sent to class GUI_Bay.

After a bay has been instantiated or edited, its outline is set/reset using either of the following two methods of class Bay: newRectangularOutline(Point2D, double, double) or newPolygonOutline(Point2D[]). Once this has been done, the various region-boundaries of the bay are set using the following methods: setPLInline(), setPLCorners(), setLLInline() and setLLMidline(). If an error is encountered during definition of the bay outline, the bay will not be added to the slab.

Column point loads, Wheel point loads, Traffic zones and UDLs

Creating/editing load objects is done in the exact same way as bay objects, however utilising different object and GUI classes, as illustrated in Table 5.1.

Table 5.1: Classes and methods involved in adding new objects to the slab

Object	**GUI class**	**Slab method**
Bay	GUI_Bay	add(Bay)
ColumnBasePL	GUI_ColumnPL	add(ColumnBasePL)
WheelPL	GUI_WheelPL	add(WheelPL)
TrafficZone	GUI_TZ	add(TrafficZone)
UDL	GUI_UDL	add(UDL)

Combined-, Dual- and Quadruple point loads

Since Combined-, Dual- and Quadruple point loads are made up of two or more single point load objects, they cannot be instantiated by the user. Instead, the user should create all single point loads independently, as described above, and then combine them using the functions provided. Methods createCombinedPLs(), createDualPLs() and createQuadruplePLs() of class Slab will determine whether or not two or more single point loads are sufficiently close together. If so, these methods will create the relevant load object(s), instances of classes CombinedPL, DualPointLoad and QuadruplePointLoad, and add them to the slab.

Line loads

LineLoad objects are instantiated, completed/edited and added to the slab inside class GUI_SlabEditor, without using a secondary GUI window. To add a line load to the slab, method addNewLL() displays a panel (see Section 6.3.3.1) which accepts load data from the user. Method addLL() then instantiates the load, assigns its attributes and adds it to the slab using the add(LineLoad) method of class Slab. If an existing line load is to be edited, the same panel is displayed by method editLL(), with the load data already filled into the correct fields.

When a line load is added to the slab, the assignCritPoints() method of class LineLoad assigns critical points to the line load. These are points where the line intersects any bay region boundaries applicable to line loads. Next, method segmentLine() creates a line load segment, an object of class LineLoadSegment, between each pair of successive critical points. In this way each line load is automatically divided into segments, each of which traverses a single bay region. Thus, the capacity of the slab to carry each segment is constant for all points along that segment. This enables a more precise and consistent technique of determining the capacity of the slab to carry a given line load.

Traffic-zone wheel-point-loads

TZwheelPL is a Subclass of WheelPL. As the name suggests, these objects represent wheel loads resulting from the presence of a traffic zone and cannot be instantiated by the user. Instead, when a traffic zone object is created, as described previously, the TrafficZone class will create one or more traffic-zone wheel-point-load(s), by the process outlined in Section 4.6. These loads are generated by methods worstComboPL1(SinglePointLoad), worstComboPL2(SinglePointLoad, double) and worstComboPL3_4(TZwheelPL, TZwheelPL, SinglePointLoad) and are added to the slab by the method createComboPLs(Slab).

Material, Fibre and Base objects

Objects of the classes Material, Fibre and Base can be instantiated by GUI_Start, GUI_SlabEditor or GUI_SlabDesign. After instantiation of a material or fibre object, the object is sent to the relevant GUI class: GUI_Material or GUI_Fibre, where its attributes can be set/edited.

Seeing as Base objects have only one attribute at this stage: modulus of subgrade reaction – *k*, a secondary editor window is not required and base objects' attributes are automatically set, during instantiation.

5.2.2 Procedure for slab-on-grade analysis

Analysis of a slab can be performed at any time, given that the slab contains at least one bay. The process of slab analysis is coordinated by the slabReport() method of class Slab. After performing various checks, calculations and comparisons, this method will return a comprehensive slab report. This report comprises several sub-sections, each compiled by a different method, providing information on a specific topic. For instance, the method limitingSinglePLToString() will generate the section of the slab-analysis report identifying the single point load(s) which limit/govern the single point load capacity of the slab. The methods which return sub-sections of the slab-analysis report are, in the order given:

- limitingSinglePLToString()
- limitingDualPLToString()
- limitingQuadPLToString()
- toStringPL()

- limitingLLSegmentsToString()
- toStringLL()
- limitingUDLToString()
- toStringUDL()

The slab-report, in the form of a String object, is then sent to class GUI_SlabAnalyse, which displays the report and enables its conversion to a PDF file.

Calculation of the various load capacities of a bay is done inside the load object classes. The abstract methods P_u(Bay) and P_p(Bay) of class PointLoading are used to calculate the bending and shear capacities of the slab: P_u and P_p respectively, considering each point load. Similarly, the bayCap() methods of classes LineLoadSegment and UDL return the relevant line load and UDL capacities of the slab. These methods regularly utilise the methods of class Bay set out in Table 5.2, which use formulae provided by TR34 (Concrete Society 2013) to obtain the values specified:

Table 5.2: Methods of class Bay which calculate bay-specific data

Bay method	**Bay value returned**
f_ctdfl()	Flexural tensile strength ($f_{ctd,fl}$)
M_un()	Negative moment capacity of plain, unreinforced concrete (M_{un})
M_u()	Ultimate positive moment capacity of reinforced concrete (M_u)
characteristic()	'Characteristic' value (λ)
l()	Radius of relative stiffness (l)
q_UDL()	UDL capacity (q)
h_c()	Crack depth of a fibre-reinforced cross-section (h_c)

5.2.3 Procedure for slab-on-grade design and optimisation

Design of the slab is initiated from class GUI_SlabEditor, after all slab attributes and loads have been set.

Firstly, class GUI_SlabDesign is employed to collect information on the type of design to be carried out. Then, depending on the type of design required, method designSlab() of this class will use a specific secondary method to coordinate the design process and compile a slab-design report.

The various design methods of class Slab are relatively simple, as their only functions are to initiate the relevant design procedures of their bays. The design methods of class Bay, however, are considerably more complex, as they are responsible for carrying out the bay-design processes set out in Sections 3.4.1 and 3.5.2.

Tables 5.3 to 5.6 outline some of the methods involved in the classes described above, to perform the four possible types of design.

Table 5.3: Basic slab design to a suitable thicknesses (classes and methods involved)

Class	**Method**
GUI_SlabDesign	basicDesign(long)
	basicDesignForH()
Slab	designSlabForH(double)
Bay	designBayForH(Slab, double)
	decreasedH(Slab)
	increasedH(Slab)

Table 5.4: Basic slab design to suitable fibre dosages (classes and methods involved)

Class	**Method**
GUI_SlabDesign	basicDesign(long)
	basicDesignForFib()
Slab	designSlabForFib(Fibre, double, double)
Bay	designBayForFib(Slab, Fibre, double, double)
	decreasedFibDos(Slab)
	slightlyDecreasedFibDos(Slab)
	increasedFibDos(Slab)

Table 5.5: Basic slab design to suitable f_{R1} & f_{R4} combinations (classes and methods involved)

Class	**Method**
GUI_SlabDesign	basicDesign(long)
	basicDesignForFib()
Slab	designSlabForFib(Fibre, double, double)
Bay	designBayForFib(Slab, Fibre, double, double)
	decreasedF_Rs(Slab)
	slightlyDecreasedF_Rs(Slab)
	increasedF_Rs(Slab)

Table 5.6: Optimised design of a slab (classes and methods involved)

Class	**Method**
GUI_SlabDesign	optimisedDesign()
Slab	optimiseSlab(Fibre, double, double, double)
Bay	optimiseBay(Slab, Fibre, double, double, double)
	designBayForH(Slab, double)
	decreasedH(Slab)
	increaseH(Slab)

After completion of the design or optimisation process, the design report compiled by one of the design methods of class GUI_SlabDesing is combined with a slab-analysis report, as discussed previously, and sent to class GUI_SlabAnalyse, which displays the report and enables its conversion to a PDF file.

Chapter 6

User interface and output

6.1 Introduction

Slab designer 2.0 is the software prototype developed to carry out a comprehensive slab analysis and design, based on physical and design data entered by a user. In order to achieve this goal, the software prototype will:

- Check if all entered values are feasible for use in analysing and/or designing a slab.
- Systematically work through the relevant part(s) of the slab-design algorithm, performing all relevant safety checks and design value calculations. In the event of an error being detected, the program operation will be aborted and the user prompted to rectify a specific faulty input value.
- Once the slab-design algorithm has been successfully worked through, the program will display the desired output values on the screen, by means of a structured report.
- The user will then have the option to save the values, print them or use them for further slab analysis or design.

This chapter serves to give an overview of the operation of the graphical user interface (GUI) of the program. This is done by carrying out a simple, hypothetical example of slab setup, analysis and design.

6.2 Welcome/initialisation window

This window initialises the program and accepts some basic data from the user – see Figure 6.1. If a slab design file with the filename extension ".slb" has been created previously, it can be opened by selecting the file and clicking "Proceed", inside the upper segment of the window.

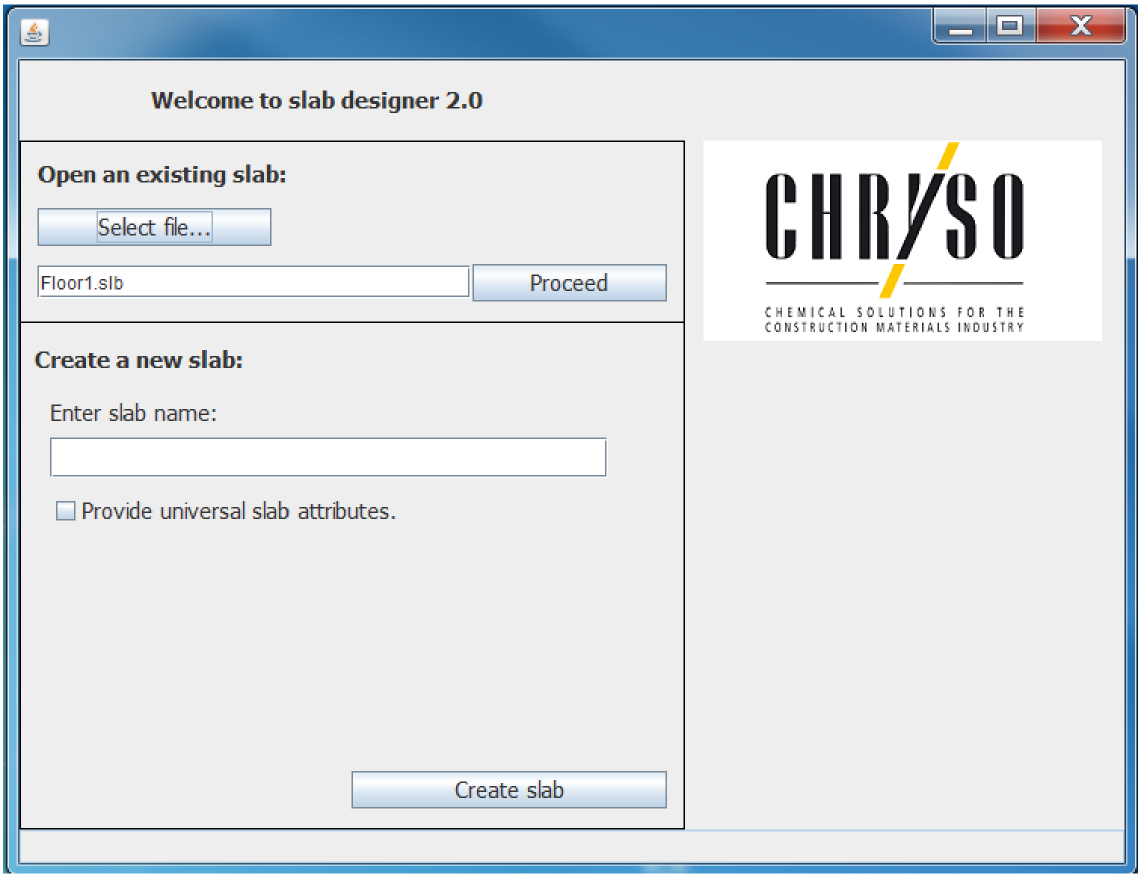


Figure 6.1: Welcoming screen, displayed after a file has been selected

Otherwise, a new slab design file should be created, within the lower segment of the window, as follows:

Inputs required:

- The user must enter a slab name.

Optional inputs:

- If the "Provide universal slab attributes" check-box is selected, additional slab info can be entered. All fields do not have to be filled in. If the slab thickness, h, modulus of subgrade reaction, k, material (see Section 6.4) and/or fibres (see Section 6.5) are known, these values can be entered to reduce repetitive entering of the same data, throughout the slab setup process.

When a new bay is added to the slab (see Section 6.6), all assigned universal slab attributes will automatically be populated into the relevant fields for that bay. The user can then choose to either utilise these standard attributes, or provide new, unique bay attributes.

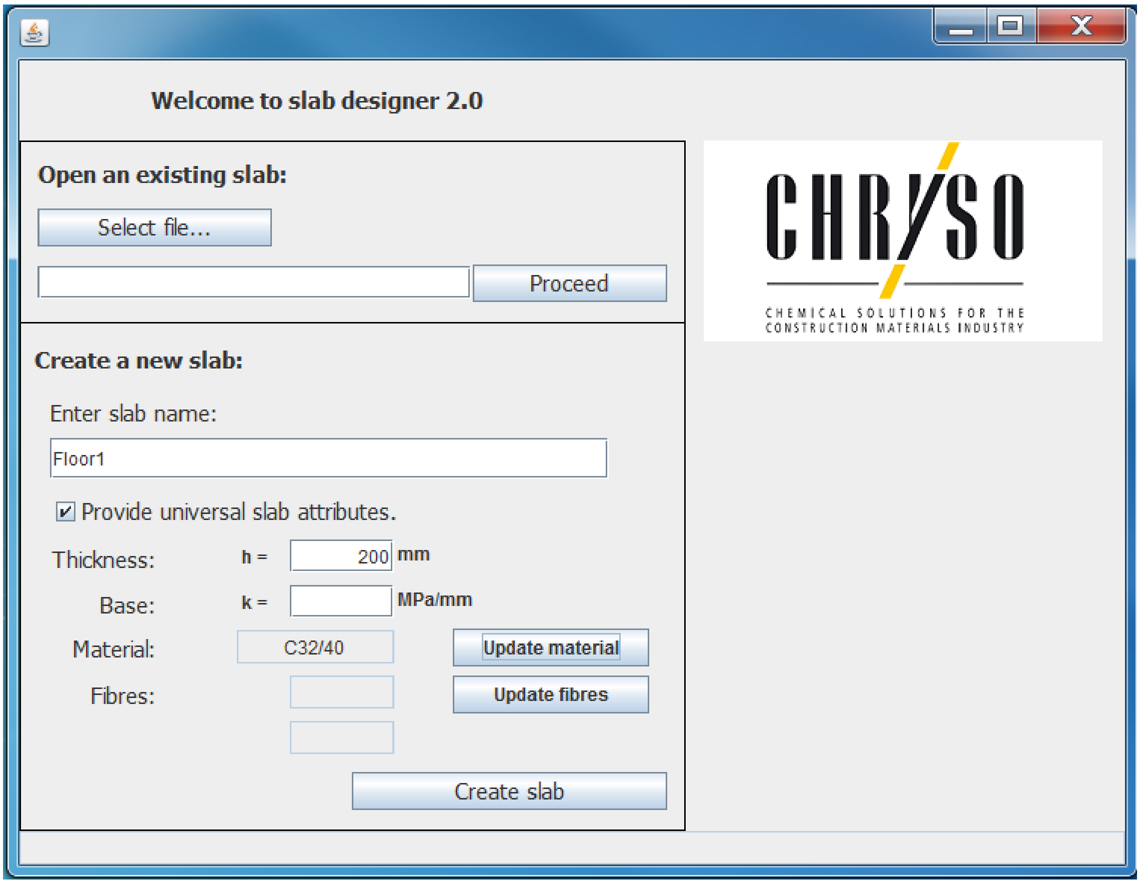


Figure 6.2: Welcoming screen with universal slab attribute fields

6.3 Slab editor window

This window acts as the central hub for slab setup, analysis and design. The window comprises of five tabs/panels, each of which handles a separate component of the slab setup, and a display panel, which shows the slab in plan, along with all loads and joints. The five tabs are individually described throughout this sub-section.

The display panel is also sensitive to clicking. Therefore, if a user clicks on the image of a certain load, zone or bay, the applicable tab will be opened, and the name of the object which was clicked on will be highlighted. Bay region information, pertaining to the selected object, will also be displayed in the communication bar at the bottom of the window.

Once the user has set up the slab properties and loadings, using the various tabs described in this section, the slab can be analysed (see Section 6.10) or designed (see Section 6.11) by selecting the relevant button at the bottom of the window.

The File menu at the top of the window can also be expanded to allow one of the following functions to be selected:

New:	Restarts the slab design process. Opens the welcome/initialisation window (Section 6.2).
Open:	Open a different pre-existing project, selected by the user.
Save:	Save the current slab file.
Save As:	Save the current slab file using a specified filename.
Rename slab:	Change the name of the current slab.

6.3.1 'Slab' tab

This tab is used to display and control the characteristics of the slab itself. Standard/universal slab attributes can be viewed and/or edited. A list of all bays on the slab is shown, along with basic bay details, and bays can be added, edited and/or removed – see Figure 6.3.

Inputs required:

Before any loads or traffic zones can be added to the slab and before the slab can be analysed or designed, at least one bay must first be added.

Optional inputs:

- Universal slab attributes can be added, removed and/or edited. The "Set attributes" button must be clicked before any changes to the standard slab attributes will take effect. If the newly entered/edited universal attributes should be applied to the pre-existing bays of the slab, the "Apply to existing bays" check-box should be selected before setting the attributes.

- Any number of bays can be added to the slab (see Section 6.6).
- Existing bays can be edited (see Section 6.6) and/or removed.

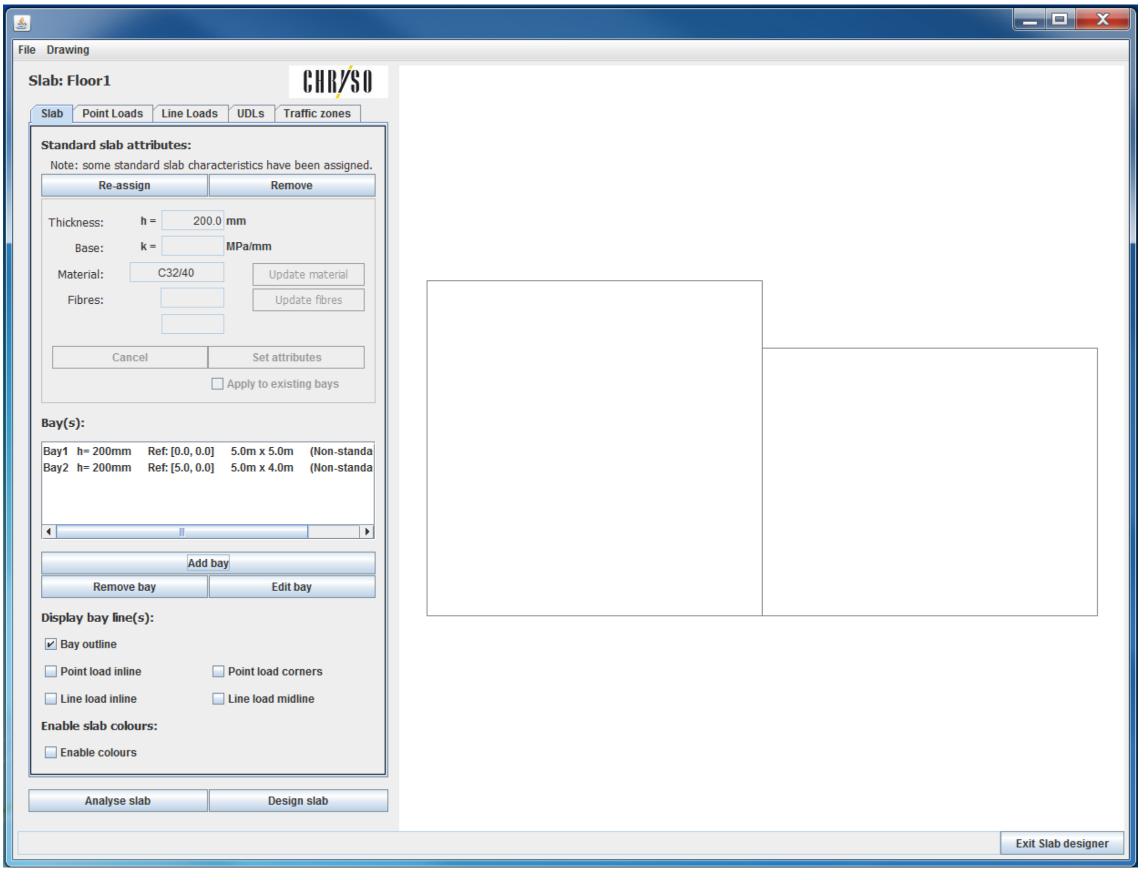


Figure 6.3: Slab tab of the Slab editor window

- Visible bay lines can be set, as desired. These lines show the different regions on the bays, which have varying load capacities. An example hereof is shown in Figure 6.4.
- If a load exceeds the capacity of the relevant bay, the user can check the zone on which the load is situated and possibly consider moving the load to a different zone.
- The slab can be displayed using basic colours by selecting the “Enable colours” check-box, as shown in Figure 6.5.

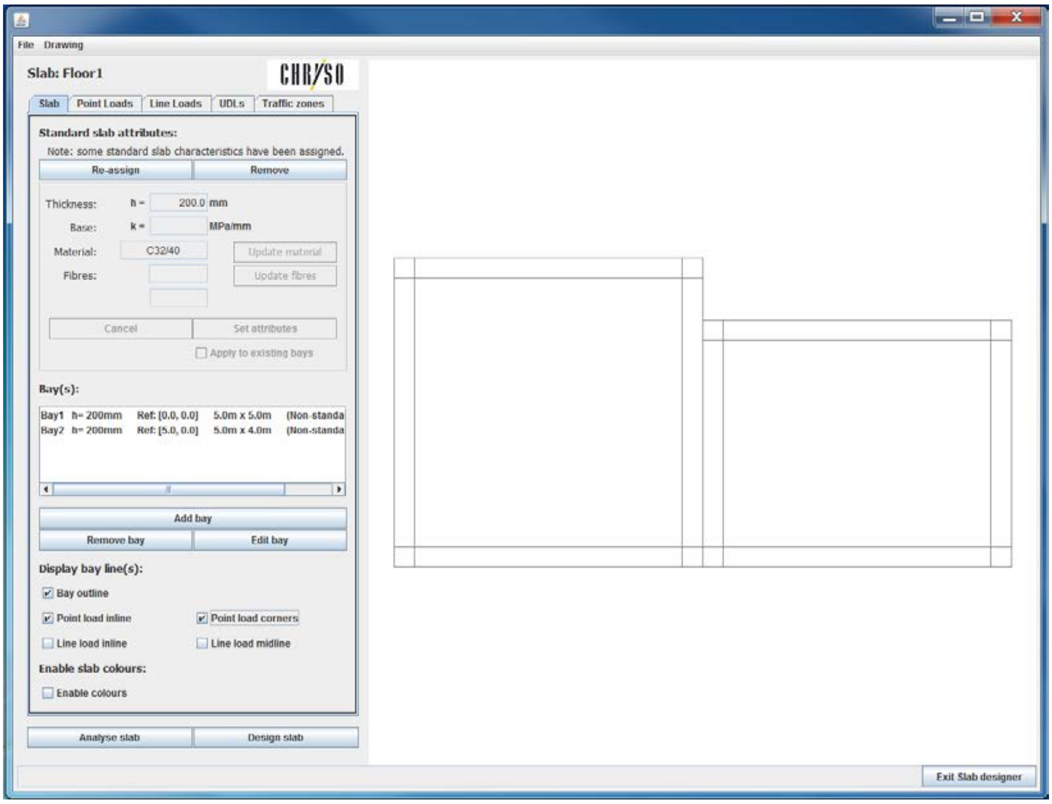


Figure 6.4: Slab tab of the Slab editor window. The internal, edge and corner zones of the bays (regarding point loads) are shown.

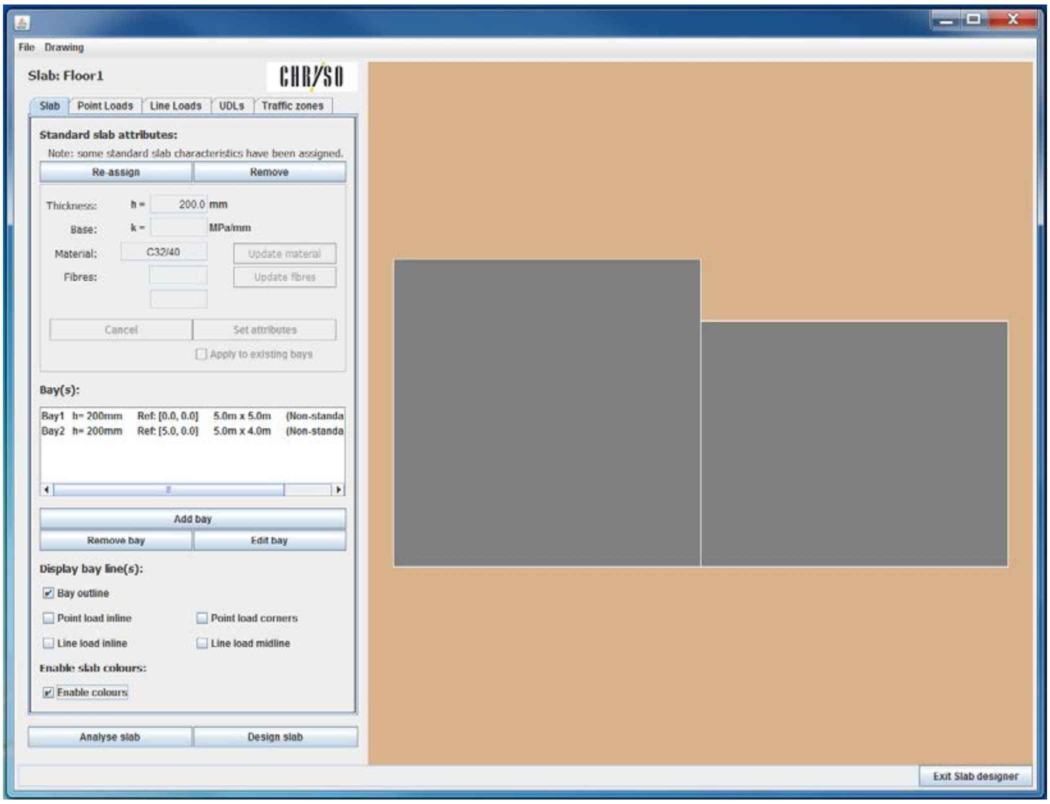


Figure 6.5: Slab tab of the Slab editor window. The slab is displayed in colours.

6.3.2 'Point loads' tab

The management of all point loads is facilitated by this tab. A list of all point loads currently on the slab is shown and any of these loads can be removed. The process of adding a new load or editing an existing one, using either of the windows described in Section 6.7, can be initiated from this tab – see Figure 6.6.

Combined-, Dual- and Quadruple point loads can be created by clicking the "Combine point loads" button, if multiple single point loads are situated close together. These loads cannot be edited, but can be separated and/or removed.

The various types of point loads are displayed using different colours, as is laid-out in the legend, at the bottom of the tab. Take note: Point loads are shown on the slab illustration by means of circular shapes with surface areas equal to the loads' effective surface areas.

Optional inputs:

- Any number of column base point loads (see Section 6.7.1) and truck wheel point loads (see Section 6.7.2) can be added to the slab.
- Existing single point loads can be edited.
- Any single or combined point load can be removed from the slab.
- If the "Combine point loads" button is clicked, relevant point loads are converted to Combined-, Dual- and Quadruple point loads, given that they are sufficiently close to each other.
- If the "Separate selected combined load" button is clicked, the selected Combined-, Dual- or Quadruple point load will be separated into its component loads. If no load has been selected or the selected load is not of an applicable type, an error message will be displayed.
- If the "Separate all combined load" button is clicked, all Combined-, Dual- and Quadruple point loads will be separated into their single point load components.

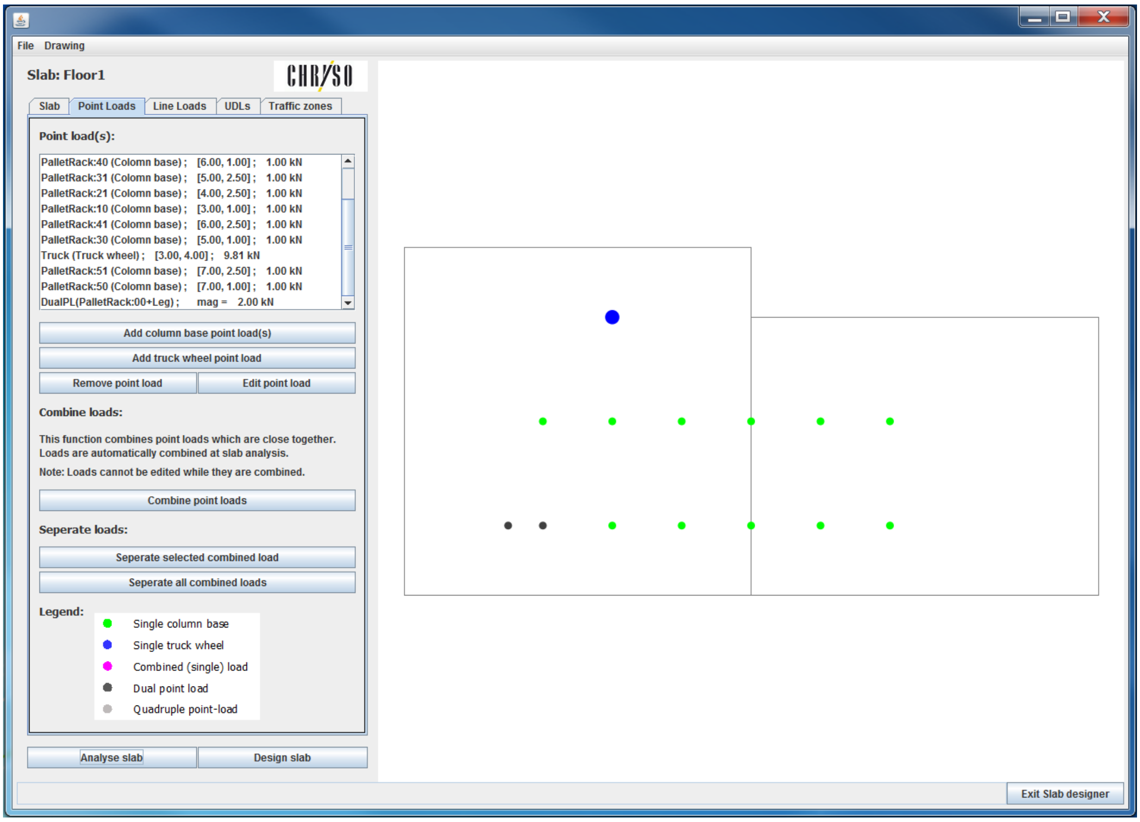


Figure 6.6: Point load tab of the Slab editor window. All point loads added are shown.

6.3.3 'Line loads' tab

The management of all line loads is facilitated by this tab. A list of all line loads on the slab is shown and loads can be added, edited and/or removed – see Figure 6.7.

Optional inputs:

- Any number of line loads can be added to the slab (see Section 6.3.3.1).
- Existing line loads can be edited and/or removed from the slab.
- When a line load is added to the slab, it is segmented according to the bay zones which it crosses. This is done through the process outlined in Section 3.6.5. These segments can be viewed by selecting the "Show line load segments" check-box (Figure 6.8). By clicking the appropriate check-boxes on the Slab tab, the user can see how the line load segments relate to the Edge-, Mid- and Internal zones of the bay(s) in terms of line loads as shown in Figure 6.9.

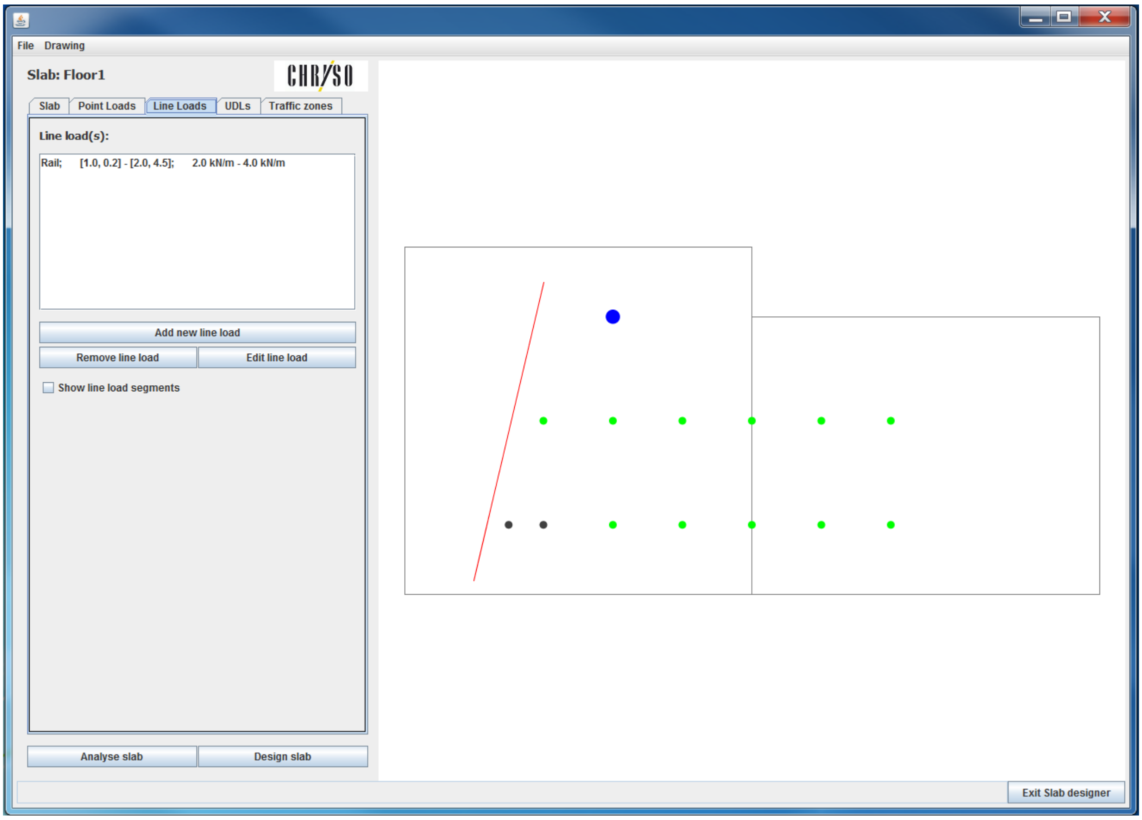


Figure 6.7: Line load tab of the Slab editor window. The line load added is shown.

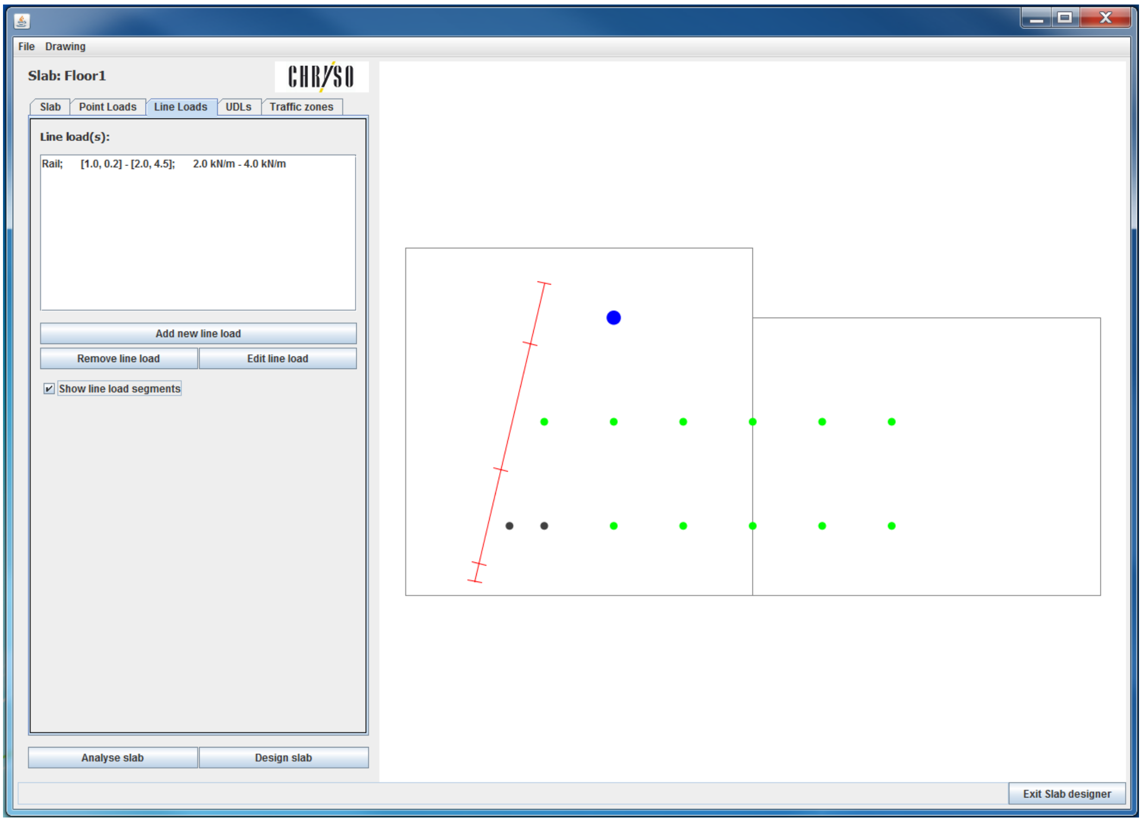


Figure 6.8: Line load segements displayed when “Show line load segments” is selected.

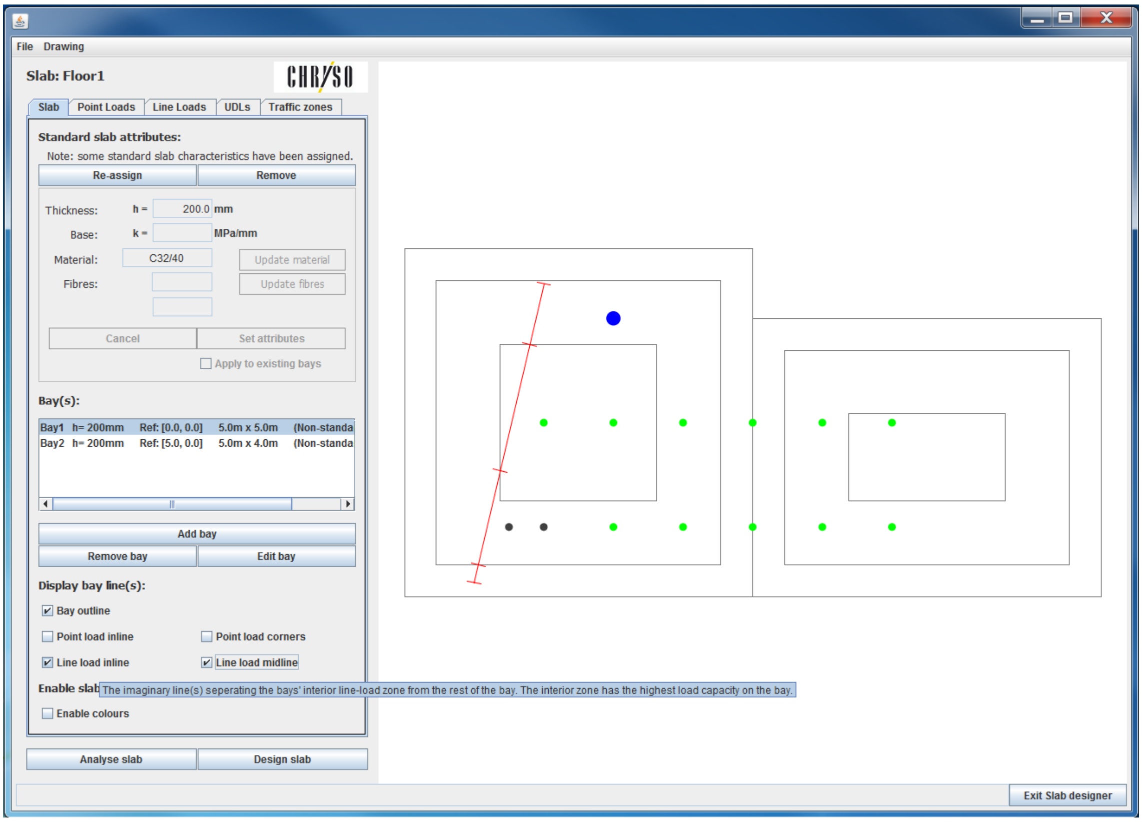


Figure 6.9: Line load segments corresponding to the edge and middle zones of the bay.

6.3.3.1 Line load editor panel

Line loads are added/edited using the Slab editor window. Thus, when the “Add new line load” button or “Edit line load” button is clicked, no new window will be opened. Instead, a panel labelled “Line load attributes” will become visible which will accept input line load data – see Figure 6.10.

Inputs required:

In order to add a new line load or edit an existing load, all data must be entered on the “Line load attributes” panel.

6.3.4 ‘Uniform distributed loads (UDLs)’ tab

The management of all uniform distributed loads is facilitated by this tab. A list of all uniform distributed loads currently on the slab is shown and any of these loads can be removed. The process of adding a new load or editing an existing one, using the window described in Section 6.8, can be initiated from this tab – see Figure 6.11.

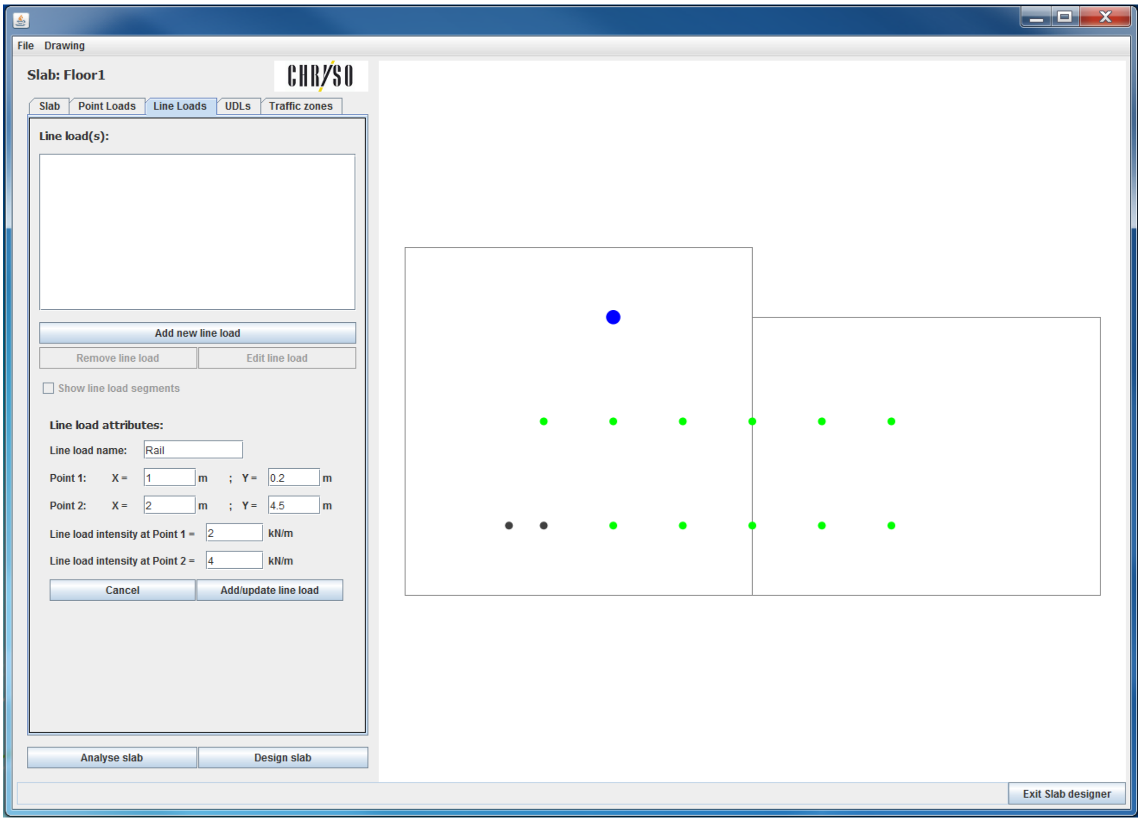


Figure 6.10: Line load editor panel on the Line load tab

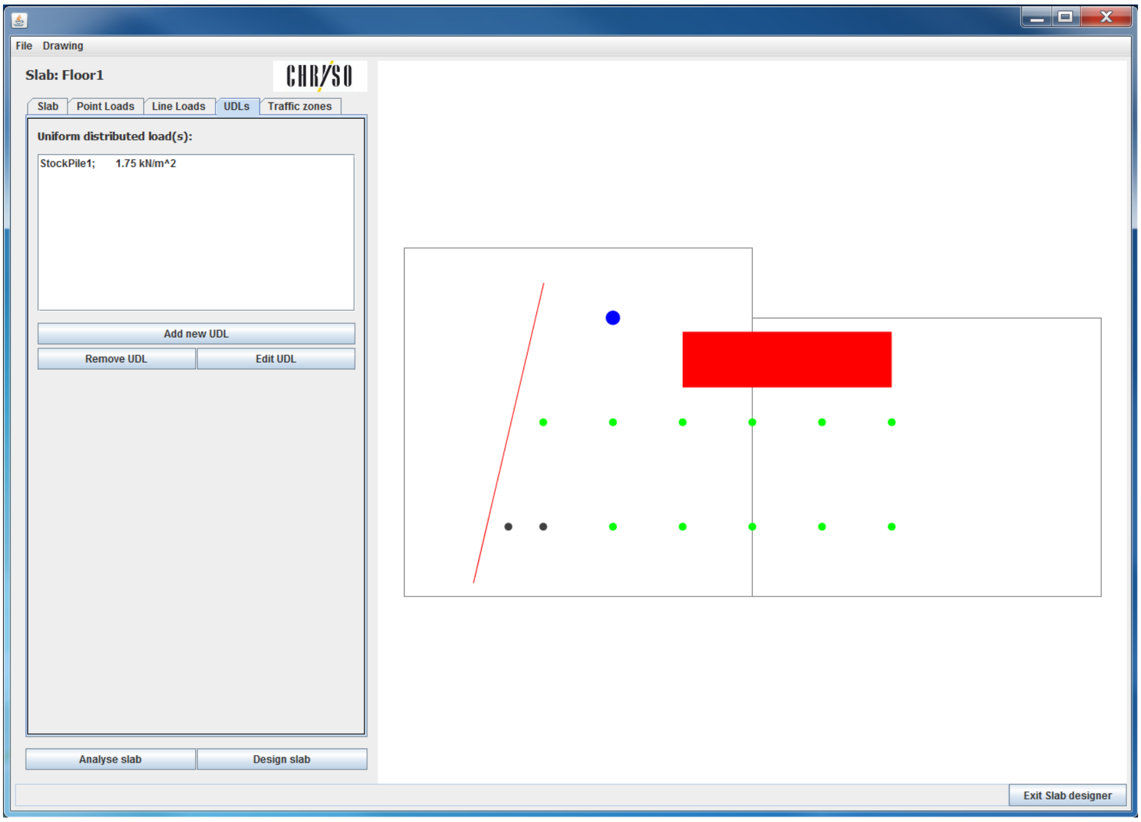


Figure 6.11: UDL tab of Slab editor window. All UDLs added are shown.

Optional inputs:

- Any number of UDLs can be added to the slab (see Section 6.8).
- Existing UDLs can be edited and/or removed.

6.3.5 'Traffic zones' tab

A traffic zone is an area on the slab which is expected to be subjected to vehicular traffic; e.g. the isle between two rows of pallet racking. After the user has designated the area, the program will create "worst case" point loads, relative to other point loads and joints on the slab, to ensure that design and analysis of the slab accounts for any and all possible loadings resulting from vehicles moving in the traffic zone. The process of determining suitable positions for the automatically generated loads is outlined in Section 4.6.

The management of all traffic zones is facilitated by this tab. A list of all traffic zones currently on the slab is shown and any of these traffic zones can be removed. The process of adding a new traffic zone or editing an existing one, using the window described in Section 6.9, can be initiated from this tab – see Figure 6.12.

Optional inputs:

- Any number of traffic zones can be added to the slab (see Section 6.9).
- Existing traffic zones can be edited and/or removed.
- If new loads are added to the slab, after a traffic zone has been defined, the "Update traffic zone loads" button can be clicked to update the "worst case loads" resulting from vehicle movement within the zone. This is done automatically before slab analysis or design takes place.

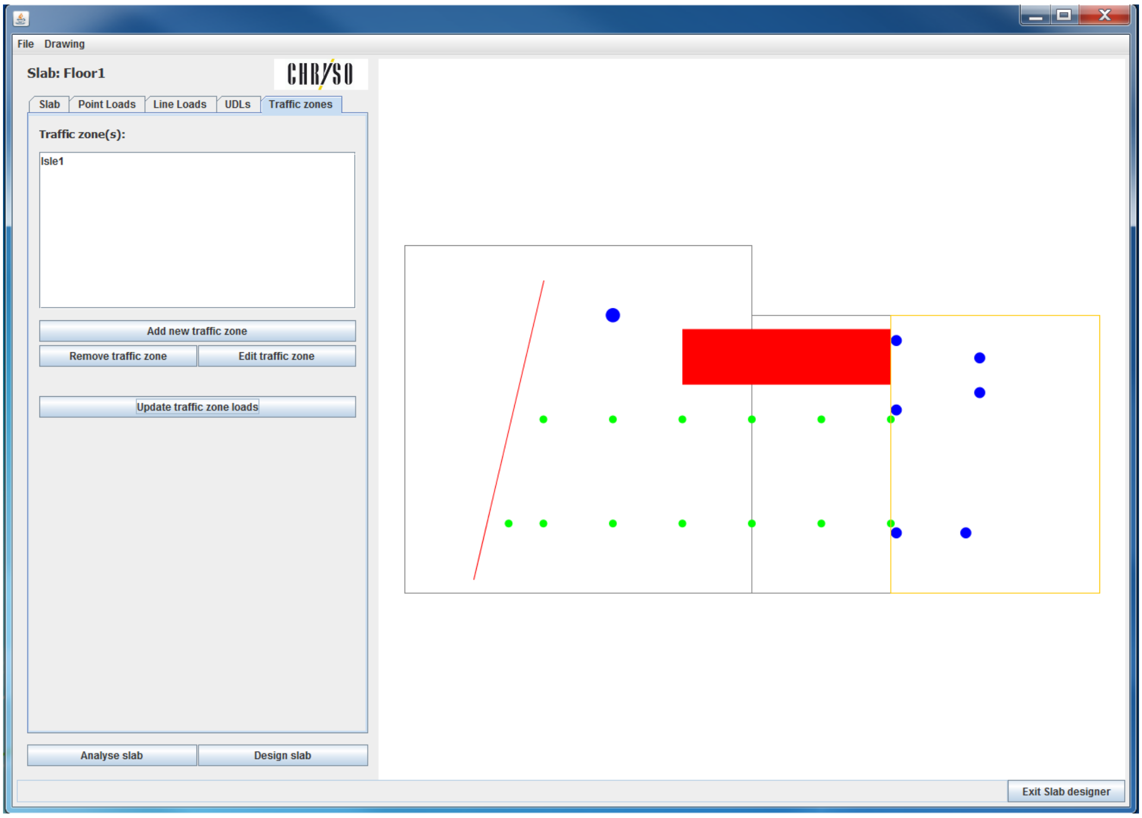


Figure 6.12: Traffic zones tab of Slab editor window. All traffic zones added are shown.

6.4 Material editor window

This window is used to enter information regarding the concrete used to construct the bays of the slab – see Figure 6.13.

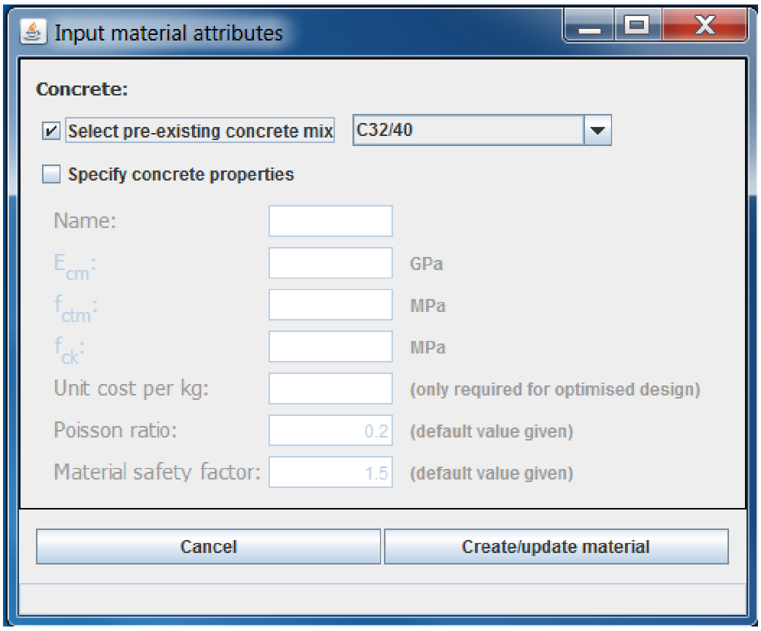


Figure 6.13: Material editor window

Inputs required:

This window allows the user to enter material information in one of two ways:

1. By selecting a pre-existing concrete mix, from a drop down list. Options listed show the respective strength classes of the pre-existing concrete mixes. The strength classes are represented by the cylinder and cube compressive strengths of the mix.
2. By manually entering specific concrete properties. Obtaining this data will typically require some laboratory testing, and allows any material to be used in slab analysis and design.

6.5 Fibre editor window

This window is used to enter information regarding the fibre reinforcing used to construct the slab – see Figure 6.14.

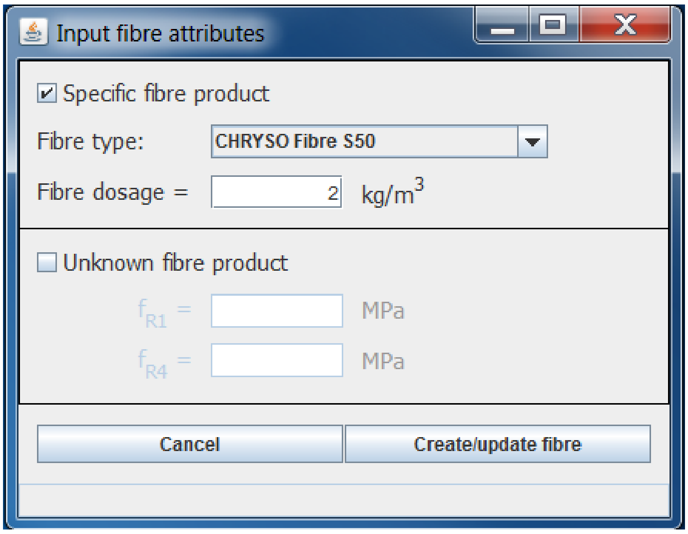


Figure 6.14: Fibre editor window

Inputs required:

This window allows the user to enter fibre information in one of two ways:

1. By selecting a pre-existing fibre type, from a drop down list, and specifying the dosage.
2. By manually entering the f_{R1} and f_{R4} values of the fibre reinforced concrete - see clause 6.3 of TR34 (Concrete Society 2013). Obtaining this data will typically require some laboratory testing, and allows any material to be used in slab analysis and design.

6.6 Bay editor window

This window is used to create new bays, or edit existing ones – see Figure 6.15.

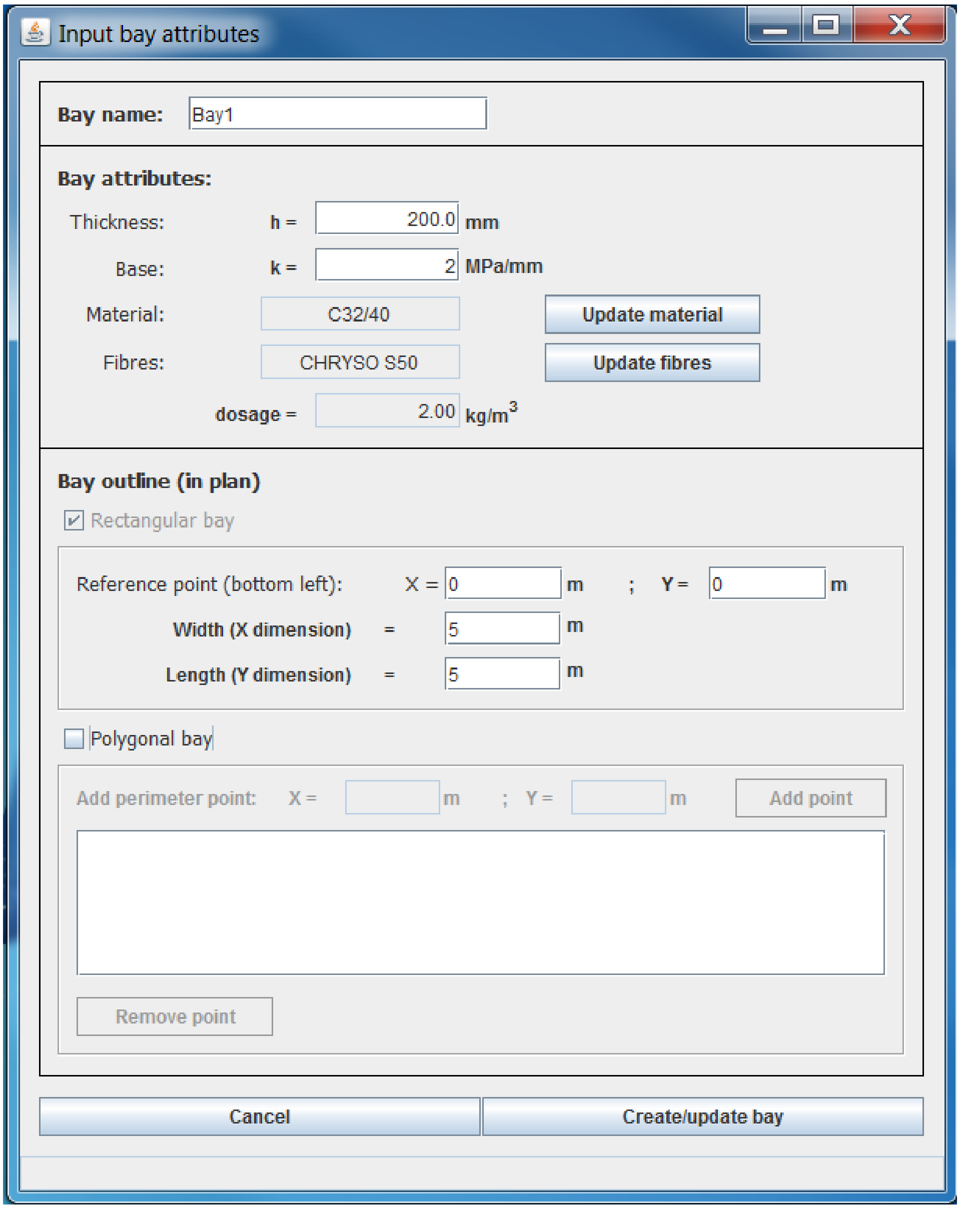


Figure 6.15: Bay editor window

Inputs required:

- All bay attributes must be entered. If any universal slab attributes have previously been set, these values will appear in the relevant fields when the window is opened.
- The bay outline must be set.
    - o If the "Polygonal bay" check-box is selected, at least three perimeter points must be added. Straight lines will be drawn between the perimeter points, to create a bay outline. The sequence of the points must be either clockwise or anti-clockwise

around the bay. All perimeter points cannot lie in a straight line, i.e. the surface area of a bay cannot equal zero.

- If the "Rectangular bay" check-box is selected, a reference point, width and length must be entered. A reference point refers to the coordinates of the bottom left-hand corner of the bay.

6.7 Point load editor windows

These windows are used to create new point loads, or edit existing ones. The window used depends on the type of point load to be created or edited.

6.7.1 Column base point load editor window

This window is used to create new column base point loads, or edit existing ones – see Figure 6.16.

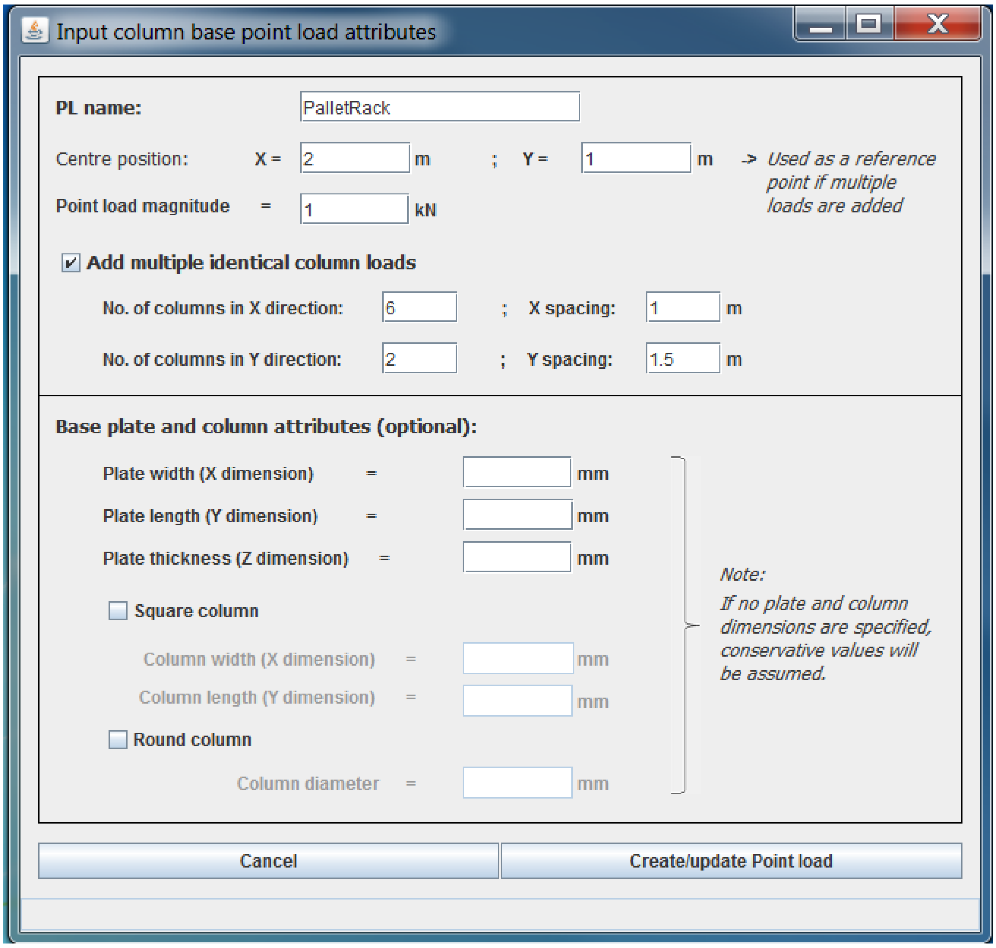


Figure 6.16: Column base point load editor window

Inputs required:

- The following data must be entered:
    - A load name
    - The coordinates of the centre of the load
    - The magnitude of the load

Optional inputs:

- If the "Add multiple identical column loads" check box is selected, the four relevant fields must be completed. In this case, several new column loads will be added to the slab, in the configuration specified by the user. The "centre position" coordinates entered is used as a reference point for the grid of new loads. This reference point refers to the coordinates of the bottom left-hand load. X and Y spacing refers to the distance between the centres of the loads in the two respective directions.
- Base plate and column dimensions can be entered for increased computational accuracy. If any of these values is omitted, conservative values for all base plate and column dimensions will be assumed.

6.7.2 Truck wheel point load editor window

This window is used to create new individual truck wheel point loads, or edit existing ones – see Figure 6.17. Note that vehicle loads are discussed in Section 6.9.

Inputs required:

- The following data must be entered:
    - A load name
    - The coordinates of the centre of the load
    - The mass of the truck
- The user must select a wheel type
    - If "Pneumatic tyres" is selected, the tyre pressure must be entered.
    - If "Solid tyres" is selected, the contact area of the wheel must be entered. The tyre manufacturer will typically have to provide this information.

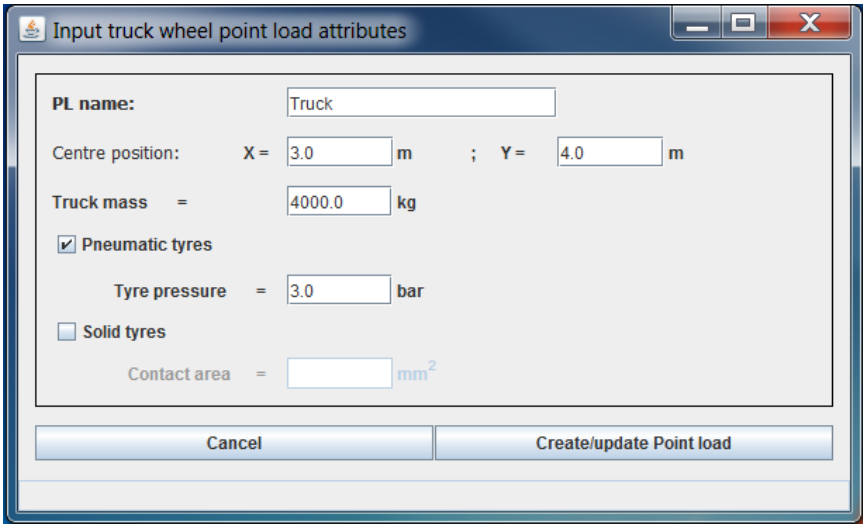


Figure 6.17: Truck wheel point load editor window

6.8 UDL editor window

This window is used to create new Uniform Distributed Loads (UDLs), or edit existing ones – see Figure 6.18.

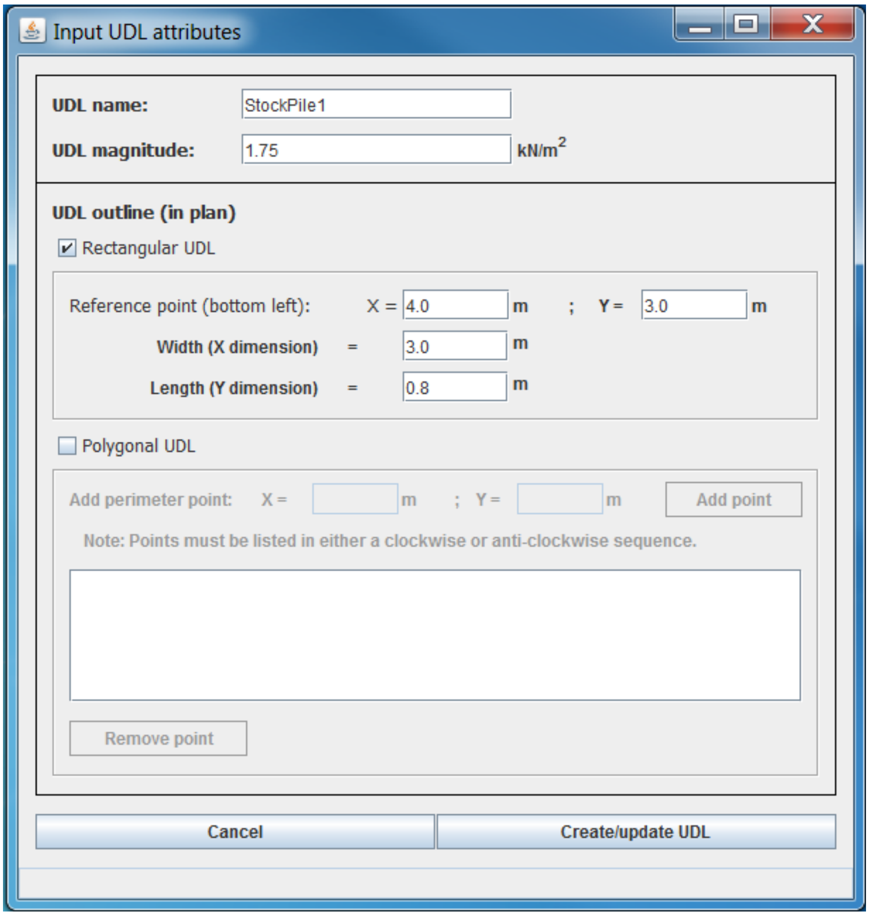


Figure 6.18: UDL editor window

Inputs required:

- All UDL attributes must be entered.
- The UDL outline must be set.
    - If the "Rectangular UDL" check-box is selected, a reference point: the coordinates of the bottom left-hand corner of the UDL, a width and a length must be entered.
    - If the "Polygonal UDL" check-box is selected, at least three perimeter points must be added. Straight lines will be drawn between the perimeter points, to create a UDL outline. The sequence of the points must be either clockwise or anti-clockwise around the UDL. All perimeter points cannot lie in a straight line, i.e. the surface area of a UDL cannot equal zero.

6.9 Traffic zone editor window

This window is used to create new traffic zones, or edit existing ones – see Figure 6.19.

Inputs required:

- The following data must be entered:
    - A traffic zone name

    Attributes regarding the relevant vehicle:

    - Vehicle mass
    - If the vehicle has pneumatic, or "air-filled" tyres, the tyre pressure must be entered.
    - If the vehicle has solid tyres, the contact area of the tyres with the slab must be entered. This information will typically be provided by the vehicle/wheel manufacturer.
    - Vehicle dimensions: front and rear axle widths and front-to-rear axle length, must be entered.
- The traffic zone outline must be set.
    - If the "Rectangular traffic zone" check-box is selected, a reference point, width and length must be entered. A reference point refers to the coordinates of the bottom left-hand corner of the traffic zone.

- If the “Polygonal traffic zone” check-box is selected, at least three perimeter points must be added. Straight lines will be drawn between the perimeter points, to create a traffic zone outline. The sequence of the points must be either clockwise or anti-clockwise around the traffic zone. All perimeter points cannot lie in a straight line, i.e. the surface area of the traffic zone cannot equal zero.

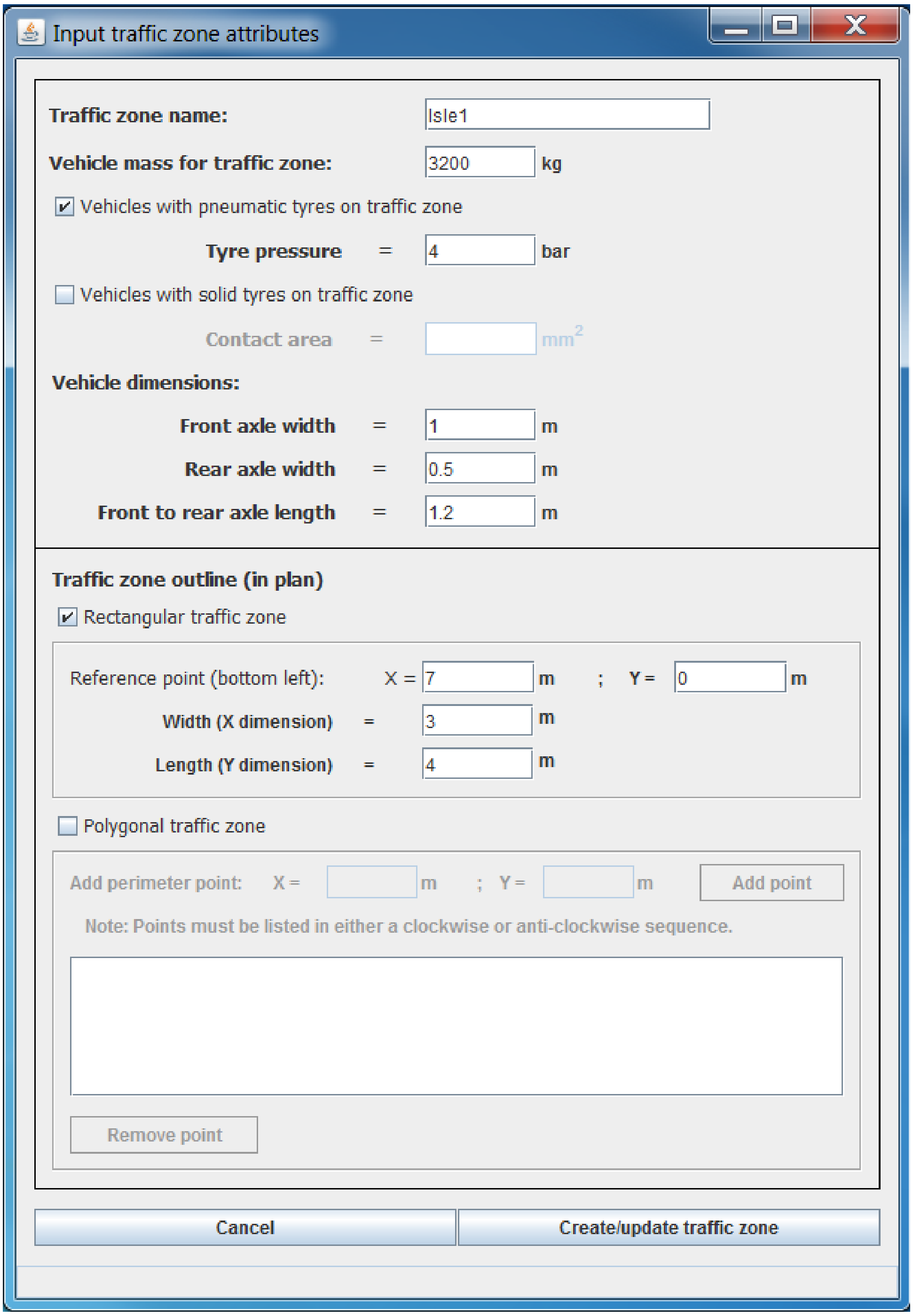


Figure 6.19: Traffic zone editor window

6.10 Slab analysis window

After all slab attributes and loads have been entered, a comprehensive slab analysis can be performed by selecting the “Analyse slab” button at the bottom of the Slab editor window. This will open the Slab analysis window. All point loads which are sufficiently close together will be converted into Combined-, Dual- or Quadruple point loads, as illustrated in Figure 6.20.

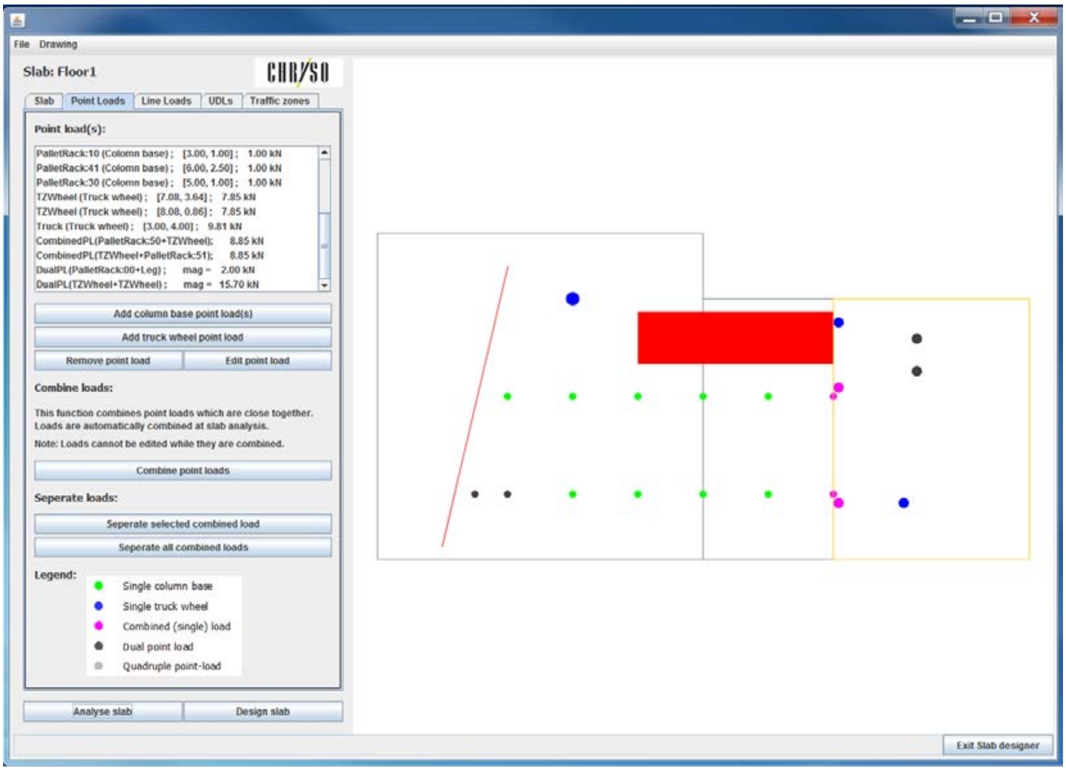


Figure 6.20: Illustration of converted point loads

A slab analysis report is compiled and displayed. This report can be exported to a PDF document, by selecting the “Create PDF document” button. See Section 4.4.1 for information on the data contained in the slab analysis report. For the example discussed in this chapter, the slab report displayed contains the following information:

- All universal slab attributes
- A list of all bays, along with their specific attributes – see Figure 6.21.
- Information regarding the limiting point loads (see Figure 6.22), including:
    - Whether or not the slab can carry all point loads on the slab
    - Which point load(s) govern the total slab point load capacity
- Specific information regarding each point load on the slab – see Figure 6.22.

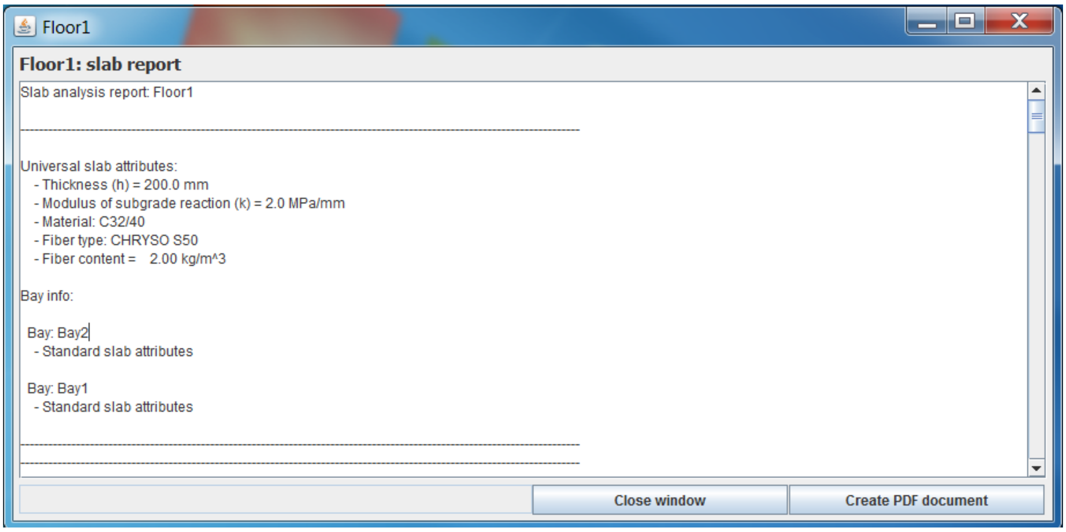


Figure 6.21: Slab analysis window – slab/bay attributes segment of slab report.

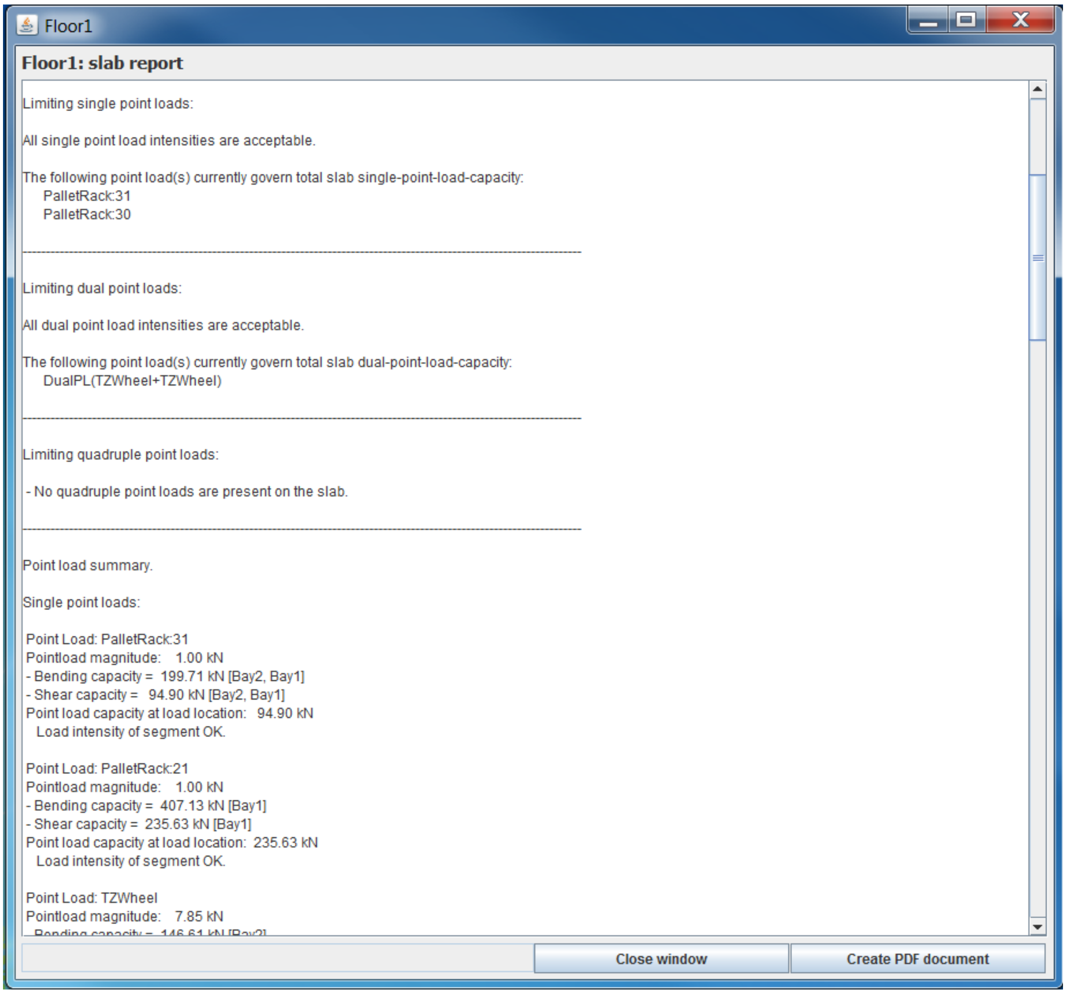


Figure 6.22: Slab analysis window – point load segment of slab report

- Information regarding the limiting line loads (see Figure 6.23), including:
    - o Whether or not the slab can carry all line loads on the slab

  - Which line load(s) govern the total slab line load capacity
- Specific information regarding each line load on the slab – see Figure 6.23.
- Information regarding the limiting UDLs (see Figure 6.24), including:
  - Whether or not the slab can carry all UDLs on the slab
  - Which UDL(s) govern the total slab UDL capacity
- Specific information regarding each UDL on the slab – see Figure 6.24.

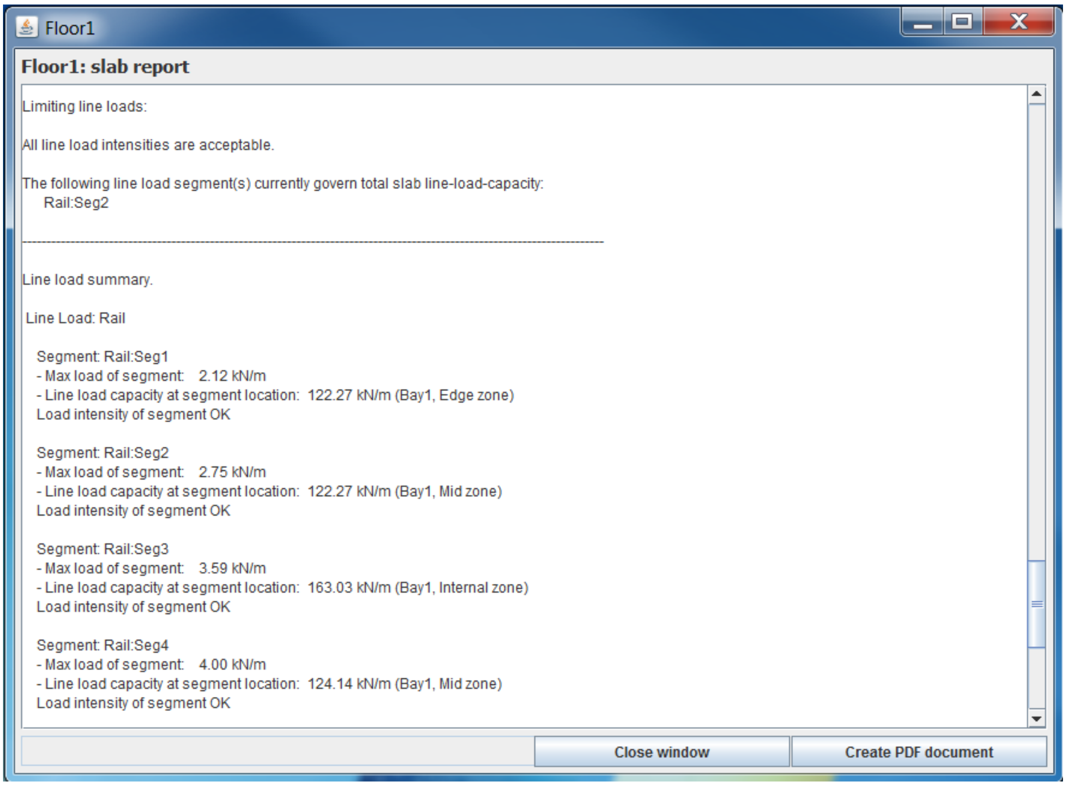


Figure 6.23: Slab analysis window – line load segment of slab report.

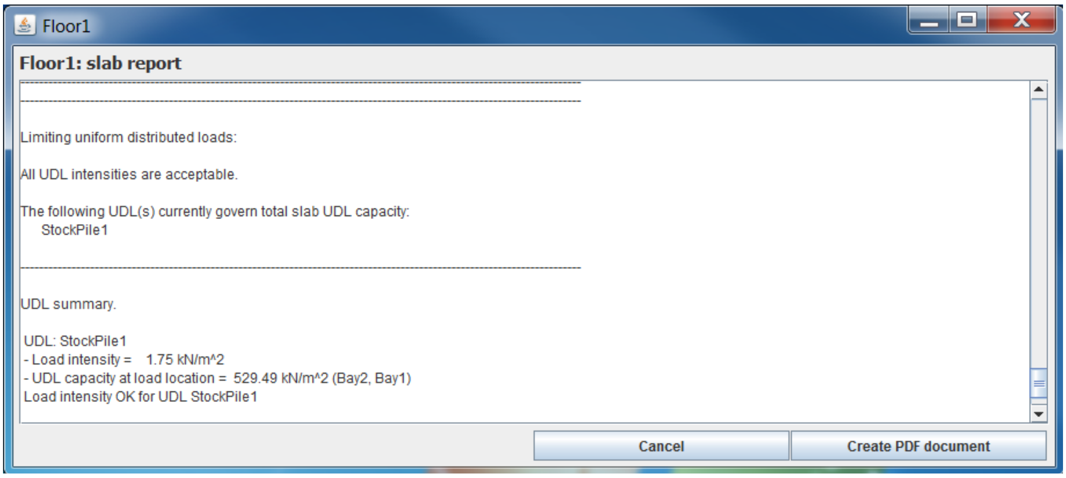


Figure 6.24: Slab analysis window – UDL segment of slab report.

6.11 Slab design window

The procedures for designing a slab are discussed in Sections 3.4.1 and 3.5.2.

This window is used to gather all required design information and initiate a specific design process, based on the objectives of the user – see Figure 6.25. Cost related calculations can be performed using any input currency symbol. The symbol "€" is used as an example in the figure below.

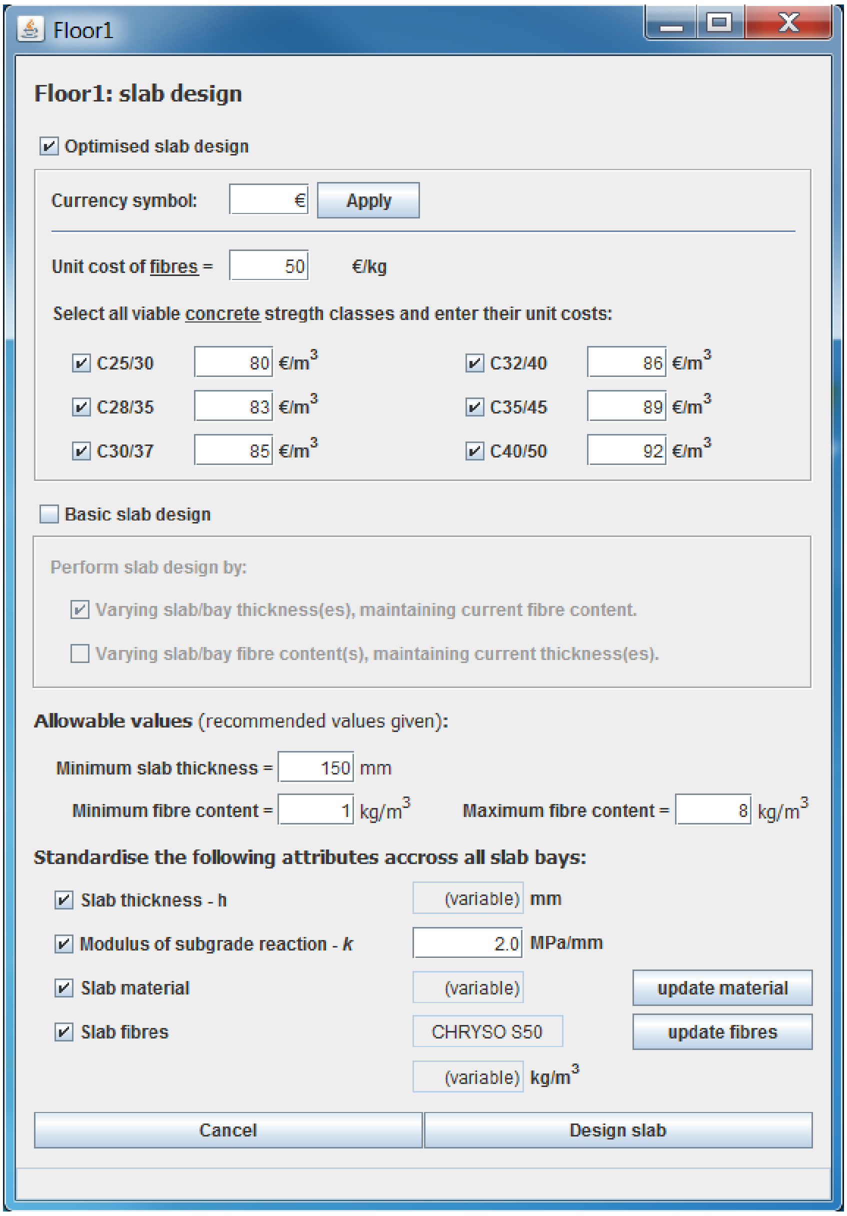


Figure 6.25: Slab design window

Slab design is initiated when the "Design slab" button, at the bottom of the Slab editor window, is selected. Once the design has finished and values have been obtained for which the slab is sufficiently strong, but not excessive, a slab analysis report is produced. This report is similar to that described in Section 6.10, but includes details of the design performed (see Figures 6.26 to 6.28).

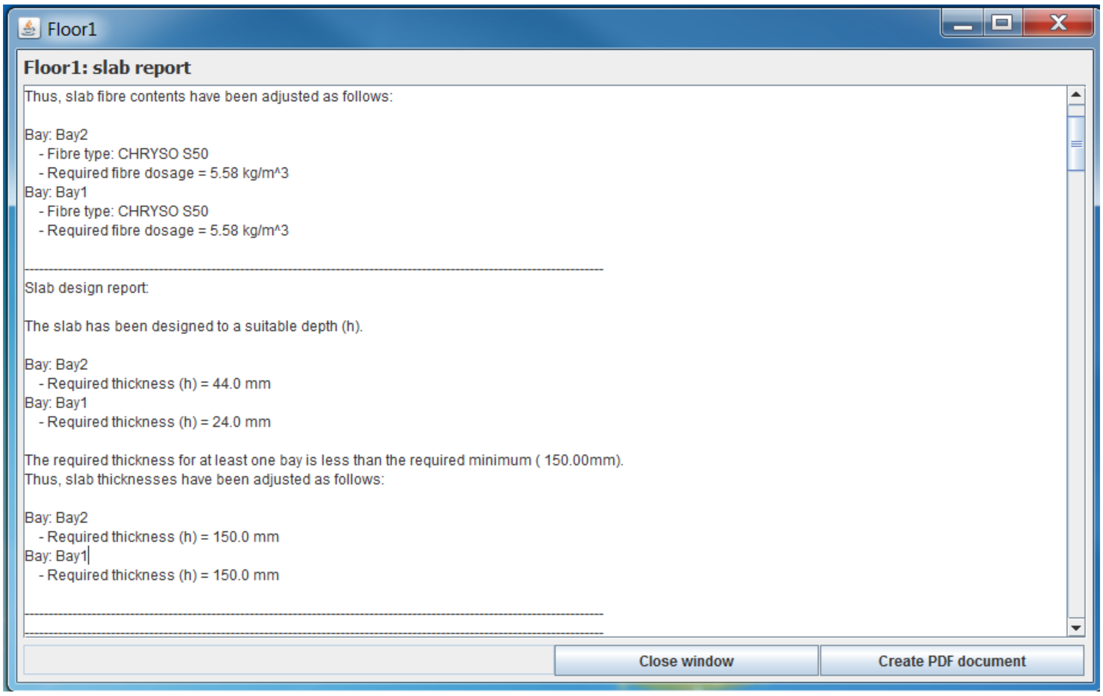


Figure 6.26: Slab analysis window – slab design based on thickness

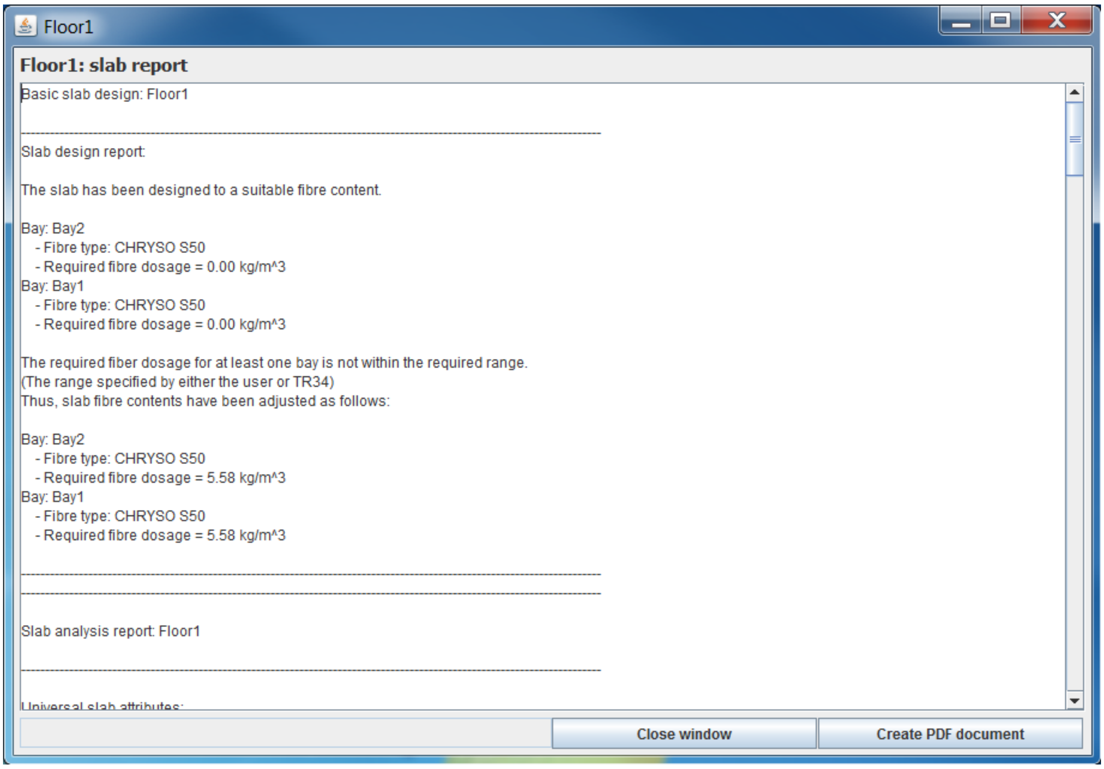


Figure 6.27: Slab analysis window – slab design based on fibre content

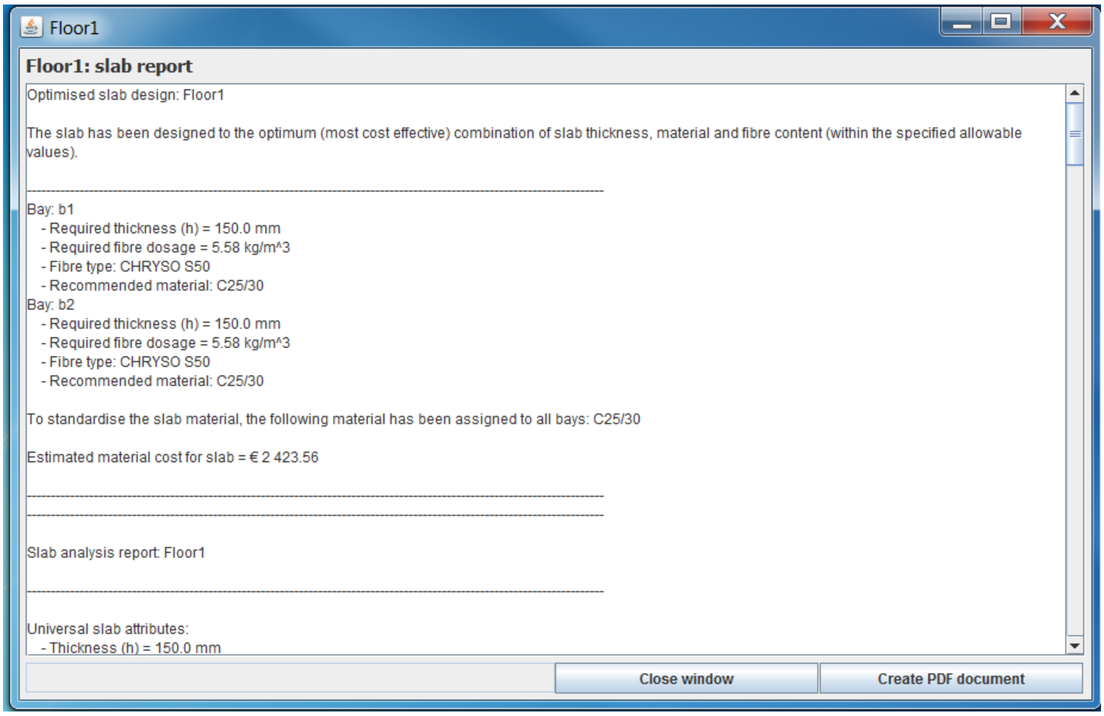


Figure 6.28: Slab analysis window – optimised slab design

Inputs required:

- The method of slab design must be selected
  - If an Optimised design is to be carried out, the unit cost of the fibres in use [per kg] must be entered. All viable concrete strength classes must be selected and their unit costs [per m^3] entered. If no viable strength classes are selected, only the materials currently assigned to the respective bays will be considered. This requires prior definition of all material unit costs.

Optional inputs:

- For optimised designs, the preferred currency can be set by entering an appropriate acronym or symbol and selecting the “Apply” button.
- If the user wishes to maintain certain standard slab attributes across all bays of the slab, these attributes can be selected. The associated value for the attribute must then be entered. All standard values are automatically set when the window is opened.
- As the value(s) for bay thickness and/or fibre content is varied during slab design, the value(s) to be varied cannot be given a pre-defined standard value. However,

these values can still be standardised across the slab, after the design has been completed and the governing value(s) have been determined.

6.12 Slab rename window

This very simple window is used to change the name of the current slab. Changing the name of the slab, however, does not change the filename by which the .slb file is saved.

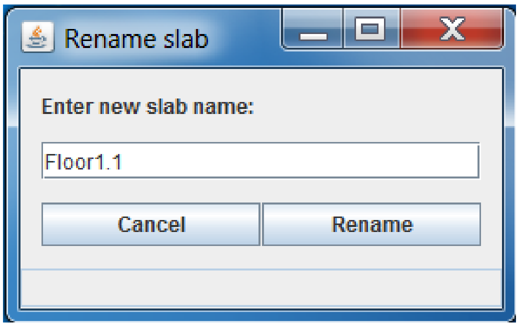


Figure 6.29: Slab rename window

Inputs required:

The user must enter a slab name and select “Rename”, or select “Cancel”.

6.13 Program exit window

If the user has not saved the current .slb file since last making changes to it, this window will prompt the user to either save the file before closing, close the file without saving or cancel closing the program.

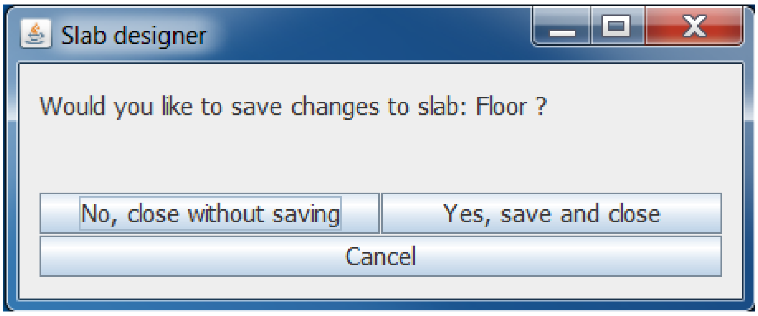


Figure 6.30: Program exit window

Chapter 7

Software testing and verification

As is the case with the development of any software package, utilising the prototype created and sensibly interpreting the results obtained, relies on the premise that the prototype functions correctly and delivers accurate results.

Correct operation of the program, considering the user interface, data collection and output format, can easily be verified through simple inspection while using the program. However, verification of the results obtained requires more rigorous scrutiny, since calculation errors and/or misinterpretations made by the program would not necessarily be obvious to the user.

This chapter serves to validate the operation of the program, as well as the results obtained, to show that slab analyses are based on accurately computed capacity values.

7.1 Unit testing of GUI components

To ensure that all user interface components of the software function correctly, a comprehensive series of unit tests have been carried out, as set out in Appendix L. The purpose of these tests is to ensure that the functionality provided by each individual component of the software is consistently and correctly executed. Functionalities are grouped according to the GUI layout set out in Chapter 6.

Since all unit tests delivered satisfactory results, it can be concluded that all GUI components of the prototype function correctly.

7.2 Verification based on hand calculations

This section compares two slab analyses, based on the same slab/load situation. The first analysis is performed by the software prototype, while the second is done by hand, using the steps and equations given in TR34 (Concrete Society 2013). This is done to verify the correctness of the various capacity and resistance values calculated by the software.

If the software prototype is capable of performing accurate slab analyses, through accurate load capacity calculations, it is assumed that the design and optimisation features of the software will also yield accurate results. This assumption is made by considering the relationships which the Design and Optimisation algorithms (see Sections 3.4 and 3.5, respectively) have with the Analysis Algorithm (see Section 3.3).

Input data for trial slab analyses:

- Slab thickness: $h = 200$ mm
- Modulus of subgrade reaction: $k = 1$ MPa/mm
- Material: C28/35 concrete
    - Compressive strength: $f_{ck} = 28$ MPa
    - Axial tensile strength: $f_{ctm} = 2.8$ MPa
    - Modulus of elasticity: $E_{cm} = 32$ GPa
- Fibre type: CHRYSO®Fibre S50
- Fibre dosage: $fD = 2$ kg/m^3
- Material safety factor: $\gamma_m = 1.5$

Analysis 1: Software prototype

In order to obtain load capacity values for a slab with the characteristics given above, a single bay is instantiated, using the software prototype. Multiple loads are then added to the various regions of the bay, as shown in Figures 7.1 and 7.2, before initialising the analysis feature of the program (see Section 4.4). Table 7.1 summarises the capacity values provided by the slab analysis report, generated by the software subsequent to slab analysis (see Section 4.4.1).

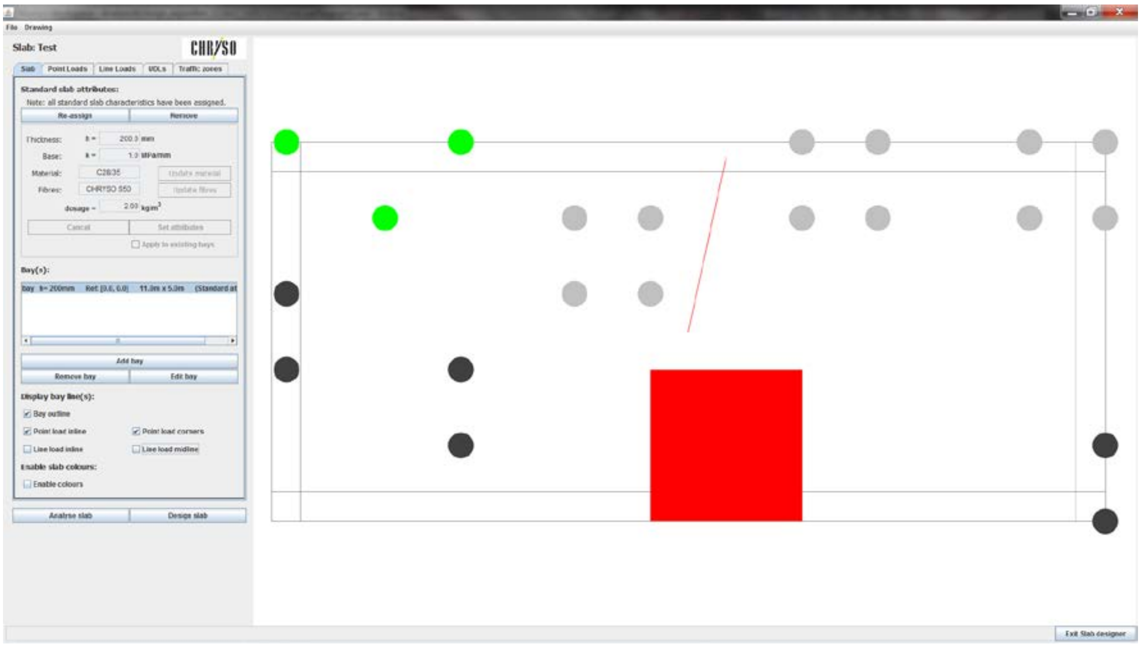


Figure 7.1: Load distribution for the trial slab, with visible point load regions

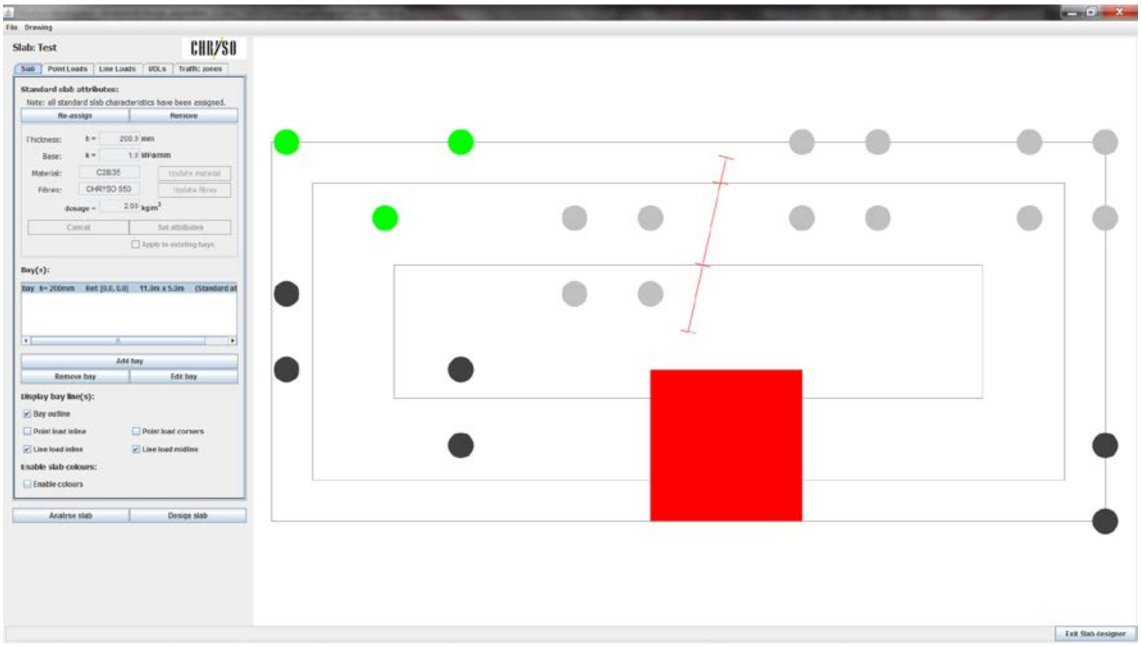


Figure 7.2: Load distribution for the trial slab, with visible line load regions and line load segments

Analysis 2: Hand calculations

By following the analysis procedure outlined in TR34, all relevant load capacity values are determined by means of hand calculations. All load values, as well as other intermediate values, are calculated accurate to two decimal places. Table 7.2 summarises the capacity values obtained through analysis of the slab by hand.

Table 7.1: Load capacity values for trial slab analysis by the software prototype

Load type	**Bay region**	**Load capacity**			**Unit**
		Bending	**Shear**	**Overall**	
Single point load	Internal zone	317.58	283.04	283.04	kN
	Edge zone	194.25	116.18	116.18	
	Corner zone	124.09	58.09	58.09	
Dual point load	Internal zone	446.43	566.08	446.43	
	Edge zone	273.06	232.36	232.36	
	Corner zone	174.43	174.27	174.27	
Quadruple point load	Internal zone	575.27	1132.16	575.27	
	Edge zone	351.87	798.44	351.87	
	Corner zone	224.77	573.50	224.77	
Line load	Internal zone	n/a		128.94	kN/m
	Middle zone			96.70	
	Edge zone			96.70	
UDL	All zones	n/a		354.86	kN/m^2

Table 7.2: Load capacity values for trial slab analysis, performed by hand

Load type	**Bay region**	**Load capacity**			**Unit**
		Bending	**Shear**	**Overall**	
Single point load	Internal zone	318.55	283.21	283.21	kN
	Edge zone	194.00	116.42	116.42	
	Corner zone	124.08	58.21	58.21	
Dual point load	Internal zone	447.79	566.42	447.79	
	Edge zone	273.06	232.29	232.29	
	Corner zone	174.42	174.35	174.35	
Quadruple point load	Internal zone	577.02	1132.16	577.02	
	Edge zone	352.39	798.70	352.39	
	Corner zone	224.75	574.03	224.75	
Line load	Internal zone	n/a		128.94	kN/m
	Middle zone			96.70	
	Edge zone			96.70	
UDL	All zones	n/a		354.78	kN/m^2

It is clear that there is very good correlation between the values obtained from Analyses 1 and 2. The largest difference between two corresponding values is a mere 0.31%, which can likely be ascribed to the fact that values are not rounded up or down during Analysis

1. Thus, it can be concluded that load capacities calculated by the software prototype are an accurate representation of the load capacities provided by TR34.

7.3 Verification based on existing slab-analysis software

Due to the fact that there is currently no commercially available software package with the ability to analyse and design fibre-reinforced slabs-on-ground, the software developed in this project could not be verified through comparison to any other existing software.

Consistent operation of the program, however, has been verified through several successful tests of the following format.

- Perform a trial design for an arbitrary slab/bay, to compute the required thickness, based on a given fibre-dosage.
- Then perform a second design, to a suitable fibre-dosage, after assigning the previously determined thickness to the bay.
- If the calculated fibre-dosage from the second design equals the input fibre dosage of the first design, consistent operation of the software can be assumed.

Chapter 8

Summary and conclusions

Through examination of all relevant sub-topics, this thesis outlines a thorough review of the theoretical concepts involving fibre-reinforced concrete slabs-on-ground. Thereafter, the process of formulating intricate algorithms and procedures, by which such slabs can be analysed, designed and optimised, is examined. Lastly, an overview is given of the software prototype developed during this study, as well as the tests carried out to verify its correct operation.

8.1 Summary of literary findings

No formal design code currently exists for the design of FRC slabs-on-ground. Thus, all influencing factors and methods of reinforcement and design are investigated in order to gain a complete understanding of the topic under consideration.

Firstly, concrete slabs-on-ground are considered, neglecting the influence of reinforcement. This provides insight into basic slab design considerations, such as surface regularity, slab serviceability and joint placement. Common loadings, along with different load locations, which slabs are often subjected to, are also investigated for the purpose of classification and quantification. Brief consideration of the soil supporting a structure provides insight into the various soil layers, methods of approximating soil behaviour and how these approximations are considered during slab design.

Secondly, fibre reinforced concrete is conceptually analysed, independent of any specific structure type. Some of the various fibre types used to strengthen concrete are considered, with emphasis placed on synthetic fibres, bearing in mind the scope of this study. The internationally preferred test method for quantifying the effects of fibres in concrete is discussed as well as practical considerations of FRC production and use. Relevant properties of SynFRC are identified along with advantages and disadvantages of its use, and existing models which predict the mechanical behaviour of MSFRC are examined.

Some common design approaches for slabs-on-ground are discussed. The yield-line theory of design is considered to be the most appropriate, considering the design guidelines in TR34 (Concrete Society 2013), and is therefore thoroughly examined.

Lastly, in order to illustrate the relevance and interactions of the preceding topics, the use of SynFRC in yield-line theory slab-on-ground design is considered.

8.2 Algorithm and procedure development summary

A hierarchical approach to creating versatile slab Analysis and Design algorithms is followed. This means that the three primary algorithms, namely analysis, design and optimisation, along with their objectives and basic layouts are set out first. Thereafter, secondary and tertiary layers of algorithms are implemented which perform more specialised functions.

The algorithms at each level will typically utilise the functionality of one or more lower level algorithms. Therefore, the most basic algorithms, at the bottom level, perform only elementary functions.

For practicality reasons, only the primary algorithms and selected lower level algorithms which carry out the following functions, are discussed in this thesis:

- Initialisation of slab attributes and bays
- Initialisation of loads
- Procedure for slab-on-grade analysis
- Procedure for slab-on-grade design
- Basic slab design to suitable thicknesses
- Basic slab design to suitable fibre dosages
- Basic slab design to suitable f_{R1} & f_{R4} combinations
- Optimised design of a slab

- Determination of bay neutral-axis and moment capacity
- Point load shear face area calculation
- Feasible fibre dosage values for slabs-on-grade
- Automated traffic-zone wheel-point-load generation

8.3 Software implementation summary

A fully functioning software prototype has been created which encapsulates all research and development performed during this project. The prototype implements all of the previously formulated algorithms and provides an effective user-interface, enabling users of all levels of expertise to perform accurate analyses, designs and optimisations of ground-supported MSFRC slabs.

8.4 Conclusions

Considering the objectives and research methodology, set out at the start of this thesis, the following conclusions can be made concerning fibre-reinforced slabs-on-ground.

- To perform accurate analyses and designs of ground-supported slabs, detailed load and bay information is required. This allows identification of the specific loads and bay regions which govern the capacity of a slab, using existing techniques. Slabs can therefore be accurately analysed and designed to thicknesses and/or fibre dosages which are sufficient, but not excessive.
- Although long term settlement and the bearing capacity of soil are known to influence any ground-supported slab, these factors are not accounted for, under the assumption that adequate ground-compaction and preparation are present. The modulus of subgrade reaction, however, is extremely relevant and this value is regularly used throughout the analysis and design processes.
- Considering the behaviour of slabs-on-ground under loading, the yield-line method of analysing slabs is considered well suited to FRC slabs. Equations derived on yield-line premises are therefore used during several capacity calculation processes. However, since TR34 adopts work based on elastic design for capacity calculations of

line- and uniform distributed load capacities, the so-called Westergaard theory for elastic structural analysis is considered highly relevant.

- By identifying clear objectives and breaking down all processes into their basic components, versatile and detailed algorithms are developed for the analysis and design of slabs-on-ground of any shape, subjected to any combination of point-, line- and/or uniformly distributed loads.
- Considering the system of physical objects and loads involved in a SOG design situation, along with the mathematical nature of most structural analysis and design problems, it has been found and demonstrated that an object model of this specific problem is feasible. An object based software package has been created using the Java programming language. It collects all necessary input data, facilitates the analysis and design processes and performs all necessary calculations.
- Following the thorough investigation of an array of optimisation techniques, it has been found that many of them could be employed to optimise the slab design process, to obtain cost-effective solutions. However, after deriving an effective objective function and evaluating the few variables involved, it was established that a simplified optimisation approach, customised to suit the design problem at hand, is most appropriate. Therefore, a unique optimisation technique is developed by which the most cost-effective combination of bay thickness and fibre dosage can be established for a specific slab/bay. This newly formulated technique is also incorporated into the developed software prototype and has been found to deliver satisfactory results.
- Through a series of basic tests, it is verified that the developed software prototype functions correctly and performs accurate computations.

8.5 Possible further research and development

The scope of the research and development described in this document is limited to what can be achieved in a single thesis. Further research and development is required to eventually develop and implement comprehensive algorithms for all aspects of ground-supported MSFRC slabs. Specific aspects that warrant attention are:

- Research on the detailing of load transfer mechanisms at slab joints and edges. This will likely result in less conservative load capacity models, especially when considering edge and corner loads.
- Investigation of the fatigue effects of heavy MHE and vehicles on ground-supported slabs. By determining the effect which long-term exposure to regular traffic has on slabs, designs can be performed which are aimed at longer serviceability periods.
- If a more accurate objective function can be derived, which takes account of all factors influencing the cost of a slab, overall optimisation of the slab can be performed. Derivation of such a function will require detailed cost information related to factors such as bay surface area, concrete modulus of elasticity, compressive and tensile strengths and fibre f_{R1} and f_{R4} values. Such optimisation will also require a more intricate technique, compared to the method employed in this thesis. Results obtained from overall optimisation may be impractical or unfeasible, from a constructability perspective.
- Stationary wheel point loads which are currently generated by traffic zone objects are placed based on simple geometric principles. Through expansion of the existing algorithms relating to traffic zones, methods can be developed by which the movement patterns and long-term effects of vehicles moving within the zone are accounted for during slab analysis and design.
- To improve user-friendliness in a professional environment, a facility can be added to the software, which allows CAD drawings of floor layouts to be imported into the program.

8.6 Concluding statement

The objectives of this study are to develop suitable design methods for FRC slabs-on-ground and to create a software package capable of performing such designs. Through examination of past research, development of versatile algorithms and creation of robust software, this thesis and the accompanying software prototype demonstrates the successful achievement of these objectives.

References

ACI, 2010. *American Concrete Institute, 360R-10, Guide to design of slabs-on-ground.*

ALANI, A.M. and BECKETT, D., 2013. Mechanical properties of a large scale synthetic fibre reinforced concrete ground slab. *Construction and Building Materials,* **41**, pp. 335-344.

ALBERTI, M.G., ENFEDAQUE, A., GALVEZ, J.C. and FERRERAS, A., 2016. Pull-out behaviour and interface critical parameters of polyolefin fibres embedded in mortar and self-compacting concrete matrixes. *Construction and Building Materials,* **112**, pp. 607-622.

AR RACKING, 2017-last update, Clad rack warehouse. Available: http://www.ar-racking.com/en/storage-systems/automated-warehouses/pallet/clad-rack-warehouses2017].

BABAFEMI, A.J. and BOSHOFF, W.P., 2017. Pull-out response of macro synthetic fibre from concrete matrix: Effect of loading rate and embedment length. *Construction and Building Materials,* **135**, pp. 590-599.

BANTHIA, N. and SAPPAKITTIPAKORN, M., 2007. Toughness enhancement in steel fiber reinforced concrete through fiber hybridization. *Cement and Concrete Research,* **37**(9), pp. 1366-1372.

BAUMANN, R.R. and WEISGERBER, F.E., 1983. YIELD-LINE ANALYSIS OF SLABS-ON-GRADE. *Journal of Structural Engineering,* **109**(7), pp. 1553-1568.

BESTER, H.L., 2017. *Generic model for predicting the performance of macro-synthetic fibre reinforced concrete for industrial flooring applications.* Stellenbosch: Stellenbosch University.

BRIDGE-END TRAINING FACILITY, 2017-last update, Industrial counterbalance forklift webpage. Available: http://www.bridgendtraining.co.uk/courses/industrial-counterbalance-forklift/2017].

BRITISH STANDARDS INSTITUTION, 2007. *EN 1997-2 Eurocode 7. Geotechnical design. Ground investigation and testing.* London: BSI.

BRITISH STANDARDS INSTITUTION, 2005. *EN 14651. Test method for metallic fibred concrete. Measuring the flexural tensile strength (limit of proportionality (LOP), residual).* London: BSI.

BRITISH STANDARDS INSTITUTION, 2004a. *EN 1992-1-1 + NA. Eurocode 2, Design of concrete structures, Part 1-1, General rules and rules for buildings.* London: BSI.

BRITISH STANDARDS INSTITUTION, 2004b. *EN 1997-1 Eurocode 7. Geotechnical design. General rules.* London: BSI.

BURATTI, N., MAZZOTTI, C. and SAVOIA, M., 2011. *Post-cracking behaviour of steel and macro-synthetic fibre-reinforced concretes.*

CONCRETE SOCIETY, 2013. *Technical report 34: Concrete industrial ground floors - a guide to design and construction, 4th edition.* Camberley: The Concrete Society.

CONCRETE SOCIETY, 2007. *Technical report 65: Guidance on the use of macro-synthetic-fibre-reinforced concrete.* Camberley: The Concrete Society.

CONCRETE SOCIETY, 2004. *The structural use of steel fabric reinforcement in ground-supported concrete floors, Project report 2.*

DI PRISCO, M., PLIZZARI, G. and VANDEWALLE, L., 2009. Fibre reinforced concrete: New design perspectives, 2009, Springer Netherlands, pp. 1261-1281.

FERRARA, L. and MEDA, A., 2006. Relationships between fibre distribution, workability and the mechanical properties of SFRC applied to precast roof elements. *Materials and Structures/Materiaux et Constructions,* **39**(288), pp. 411-420.

GOODCHILD, C.H., 2000. Rationalisation of flat slab reinforcement. Crowthorne: British Cement Association, pp. 210 pp.

HETÉNYI, M., 1971. *Beams on elastic foundation: theory with applications in the fields of civil and mechanical engineering.* University of Michigan.

INDIAMART, 2017-last update, Stacker cranes. Available: https://dir.indiamart.com/impcat/stacker-cranes.html2017].

INDUSTRY PLAZA, 2017-last update, Side fork lift truck. Available: https://www.industry-plaza.com/side-fork-lift-truck-o5624.html2017].

JAMES, I., GOPALARATNAM, V. and GALINAT, M., 2002. State-of-the-art report on fibre reinforced concrete. *Manual Concrete Practice,* **21**, pp. 2-66.

KENNEDY, G. and GOODCHILD, C.H., 2004. *Practical yield line design.* Surrey: The Concrete Centre.

KROST SHELVING AND RACKING, 2017-last update, Advantages of considering cantilever racking. Available: http://krostshelving.com/news/advantages-of-considering-cantilever-shelving-04052014/2017].

LOSBERG, A., 1978. PAVEMENTS AND SLABS ON GRADE WITH STRUCTURALLY ACTIVE REINFORCEMENT. **75**(12), pp. 647-657.

MECALUX, 2017a-last update, Mobile pallet racking Movirack. Available: https://www.mecalux.com/pallet-racks/mobile-pallet-racking-movirack2017].

MECALUX, 2017b-last update, Push-back pallet racking. Available: https://www.mecalux.com/pallet-racks/push-back-pallet-racking2017].

MENTOR TRAINING, 2017-last update, Fork lift truck training. Available: https://www.mentorflttraining.co.uk/courses/articulated-trucks2017].

MEYERHOF, G.G., 1962. Load-carrying capacity of concrete pavements. *ASCE -- Proceedings -- Journal of the Soil Mechanics and Foundations Division,* **88**, pp. 89-116.

NIEUWOUDT, P.D. and BOSHOFF, W.P., 2017. Time-dependent pull-out behaviour of hooked-end steel fibres in concrete. *Cement and Concrete Composites,* **79**, pp. 133-147.

NILKAMAL, 2017-last update, Pallet live storage systems. Available: http://www.nbss.in/palletlivestorage.htm2017].

PALLETTRUCK WAREHOUSE, 2017-last update, Standard pallet truck product catalogue. Available: http://pallettruckwarehouse.co.uk/standard-pallet-trucks/1-euro-pallet-truck-pt-04-25t-1150mm-x-550mm.html2017].

RAYMOND HANDLING CONCEPTS, 2017-last update, What is a reach truck and why do I need one?. Available: https://raymondhandling.com/learn/library/what-is-a-reach-truck-and-why-do-i-need-one/2017].

RILEM, 2002. Final recommendations of TC 162-TDF. Test and design methods for steel fibre reinforced concrete: bending test. *Materials and Structures,* **35**, pp. 579.

SOLYOM, S. and BALAZS, G.L., 2016. Influence of FRC on bond characteristics of FRP reinforcement, *11th fib International PhD Symposium in Civil Engineering, FIB*

2016, August 29, 2016 - August 31, 2016 2016, A.A. Balkema Publishers, pp. 271-278.

TRANPAK, 2017-last update, Drive-in racking for pallets. Available: http://www.tranpak.com/tools/racking-styles/drive-in2017].

WALRAVEN, J.C., 2009. High performance fiber reinforced concrete: Progress in knowledge and design codes, 2009, Springer Netherlands, pp. 1247-1260.

WESTERGAARD, H.M., 1926. Stresses in concrete pavements computed by theoretical analysis. *United States Department of Agriculture -- Public Roads,* **7**(2), pp. 25-35.

WESTERGAARD, H.M., 1925. Computation of stresses in concrete roads, *The fifth annual meeting of the Highway Research Board, Vol 5, Part 1* 1925, pp. 90-112.

WESTERGAARD, H.M., 1923. *On the design of slabs on elastic foundation with special reference to stresses in concrete pavements.* Copenhagen (in German): Ingenioren.

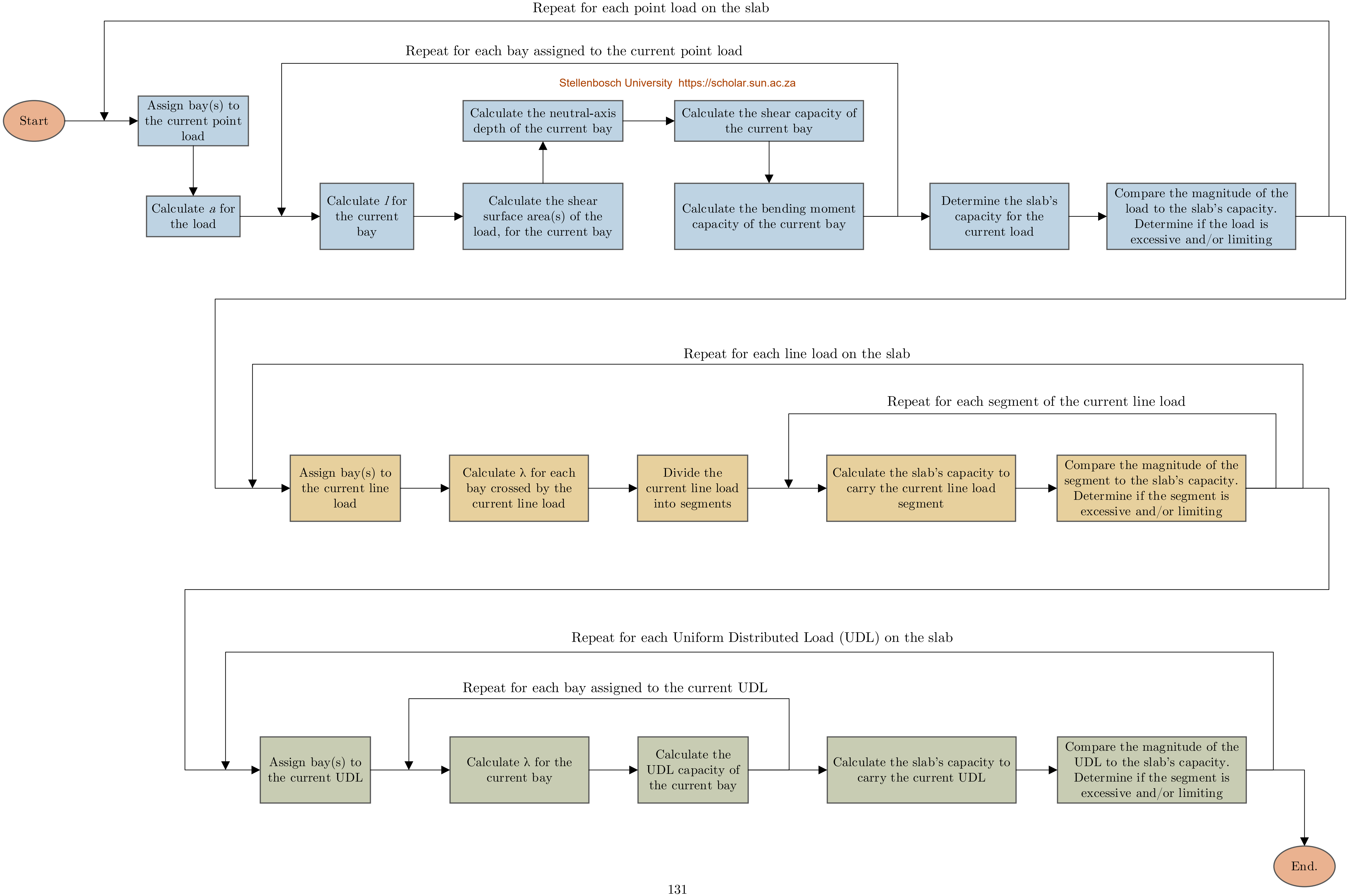
WINKLER, E., 1867. Die lehre von der Elastizitat und Festigkeit. *Verlag,* , pp. 182.

ZHAO, Q., YU, J., GENG, G., JIANG, J. and LIU, X., 2016. Effect of fiber types on creep behavior of concrete. *Construction and Building Materials,* **105**, pp. 416-422.

Appendix A

Flowchart for the slab analysis procedure

The flowchart on the following page illustrates the steps performed during the slab analysis procedure, as outlined in Section 3.3.1.

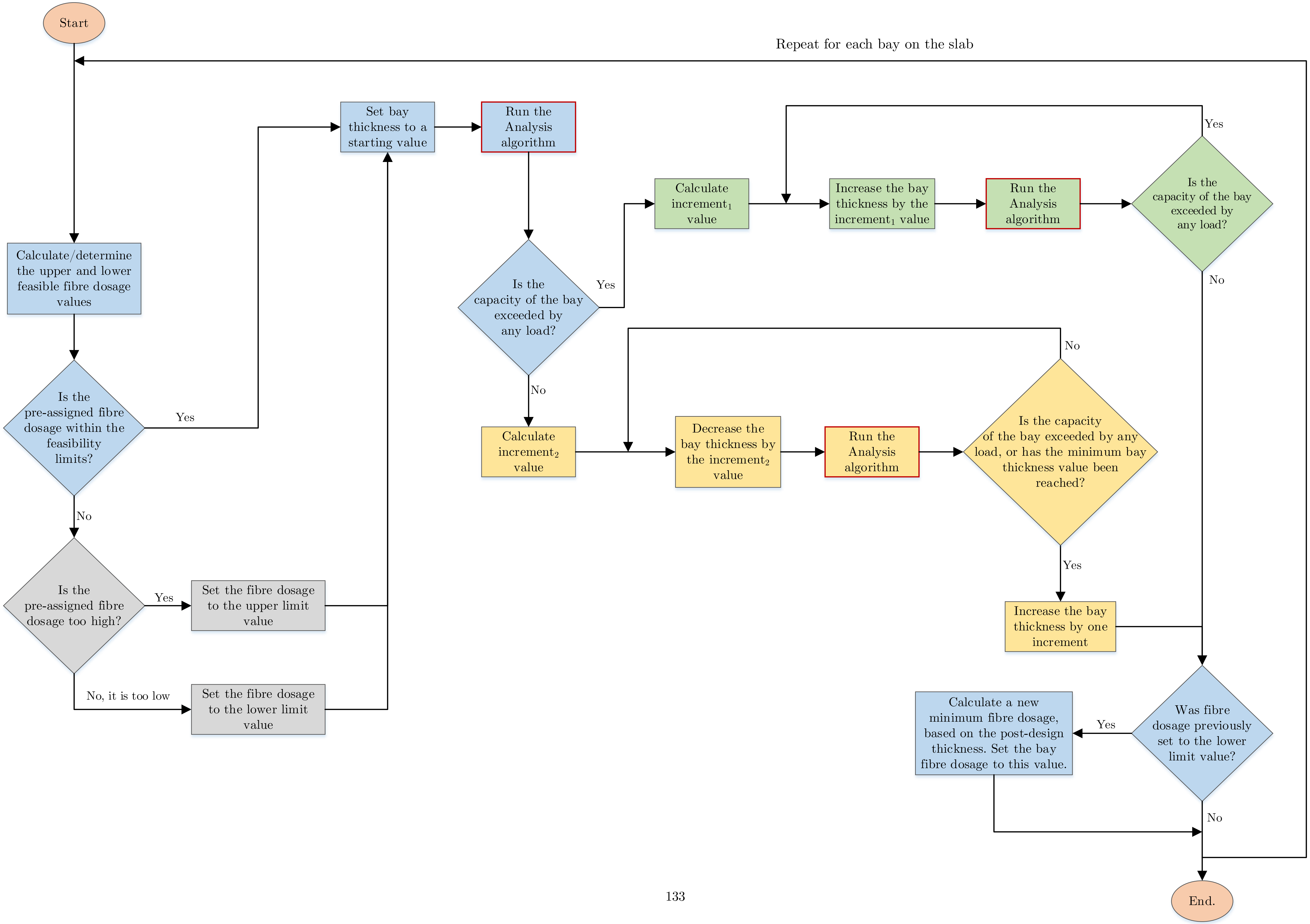


Stellenbosch University https://scholar.sun.ac.za

Appendix B

Flowchart for the basic design procedure to calculate suitable thicknesses for a slab-on-grade

The flowchart on the following page illustrates the steps performed during the basic slab design procedure, to calculate suitable bay thicknesses, as outlined in Section 3.4.1.1.

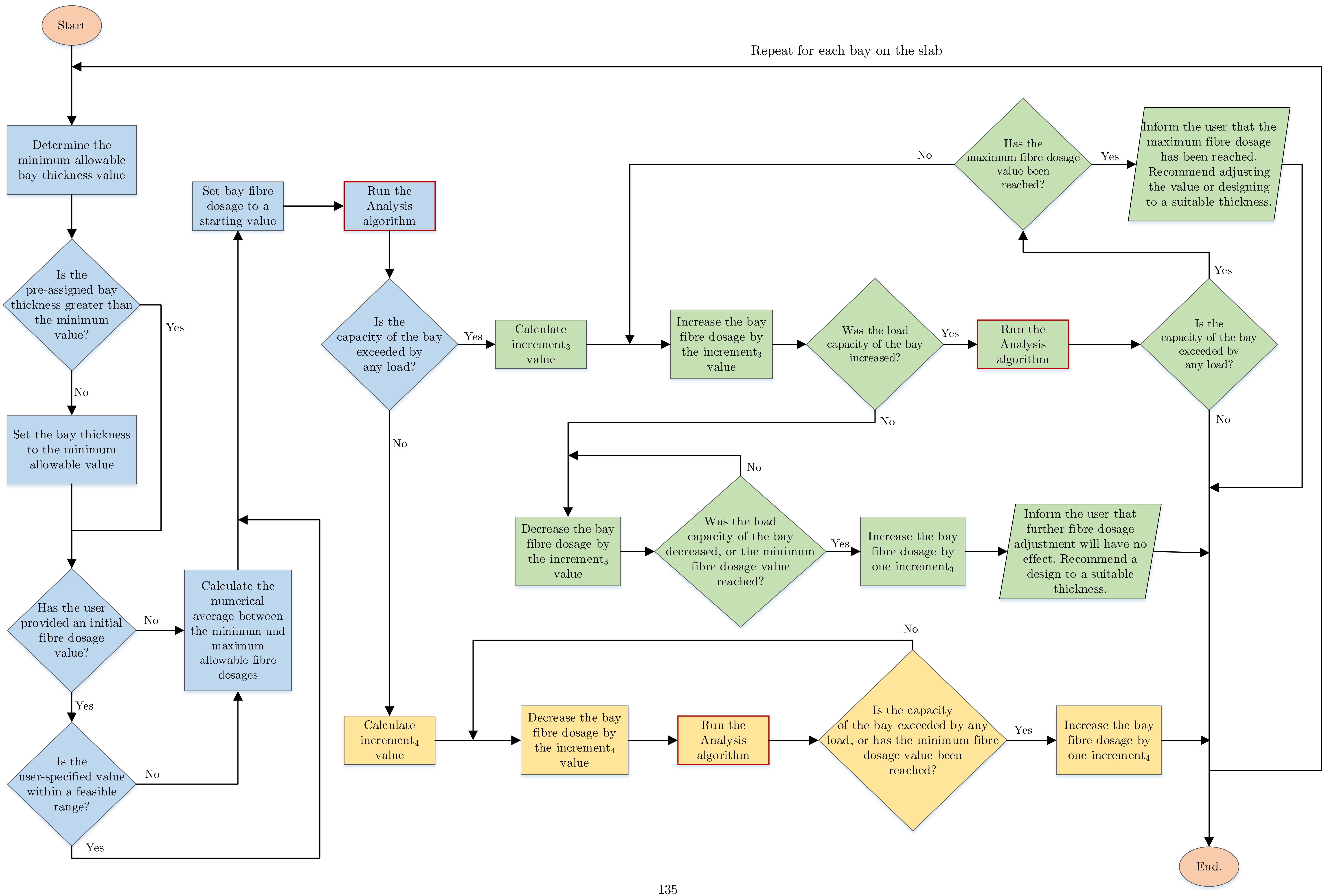

Start
Repeat for each bay on the slab
Calculate/determine the upper and lower feasible fibre dosage values
Is the pre-assigned fibre dosage within the feasibility limits?
Yes
No
Is the pre-assigned fibre dosage too high?
Yes
No, it is too low
Set the fibre dosage to the upper limit value
Set the fibre dosage to the lower limit value
Set bay thickness to a starting value
Run the Analysis algorithm
Is the capacity of the bay exceeded by any load?
Yes
No
Calculate increment$_1$ value
Increase the bay thickness by the increment$_1$ value
Run the Analysis algorithm
Is the capacity of the bay exceeded by any load?
Yes
No
Calculate increment$_2$ value
Decrease the bay thickness by the increment$_2$ value
Run the Analysis algorithm
Is the capacity of the bay exceeded by any load, or has the minimum bay thickness value been reached?
No
Yes
Increase the bay thickness by one increment
Was fibre dosage previously set to the lower limit value?
Yes
No
Calculate a new minimum fibre dosage, based on the post-design thickness. Set the bay fibre dosage to this value.
End.

Appendix C

Flowchart for the basic design procedure to calculate suitable fibre dosages for a slab-on-grade

The flowchart on the following page illustrates the steps performed during the basic slab design procedure, to calculate suitable bay fibre dosages, as outlined in Section 3.4.1.2.

Appendix C: Basic slab design procedure, to suitable fibre dosages - flowchart

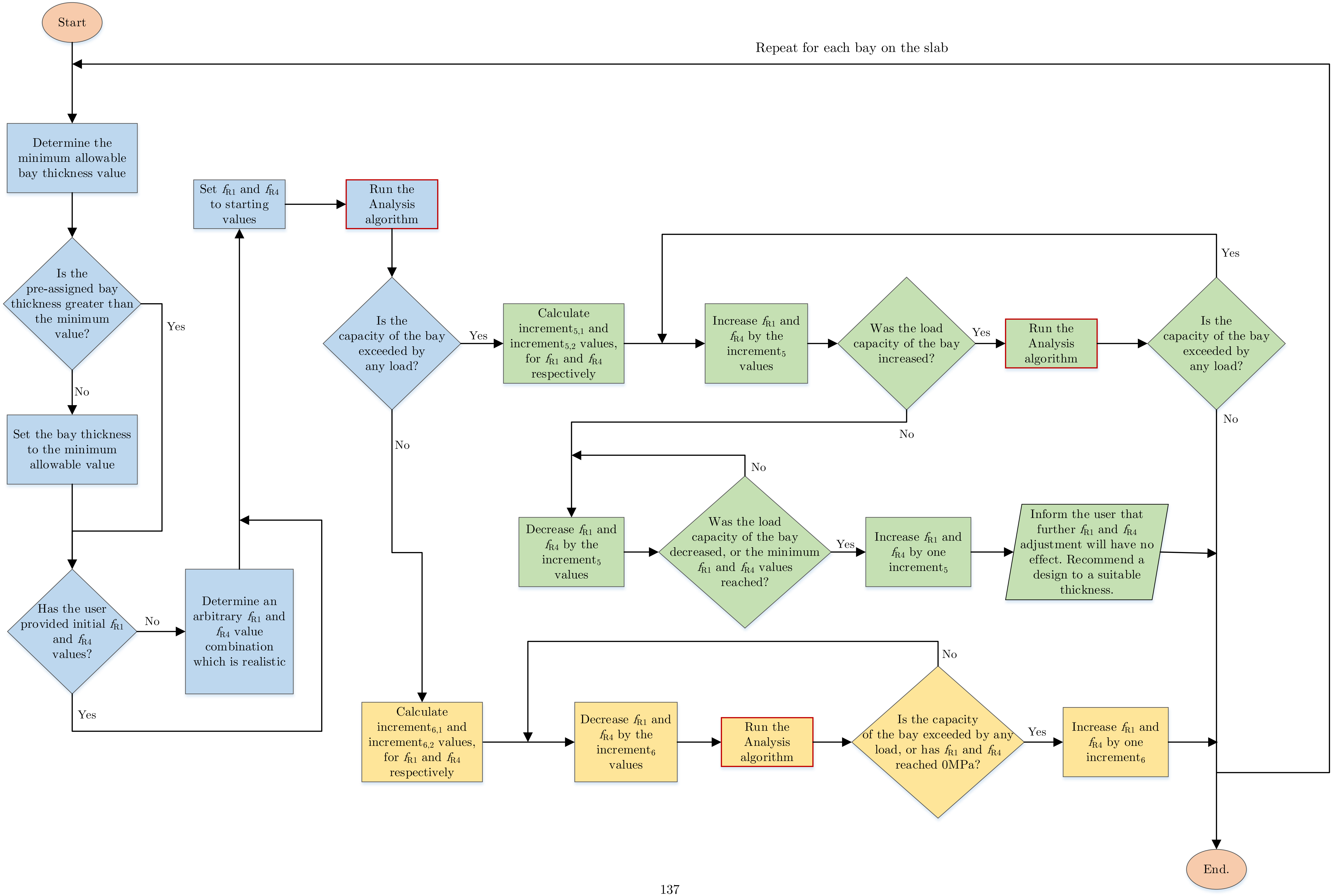


Appendix D

Flowchart for the basic design procedure to calculate suitable f_{R1} and f_{R4} combinations for a slab-on-grade

The flowchart on the following page illustrates the steps performed during the basic slab design procedure, to calculate suitable bay f_{R1} and f_{R4} combinations, as outlined in Section 3.4.1.3.

Appendix D: Basic slab design procedure, to suitable f_{R1} and f_{R4} combinations - flowchart

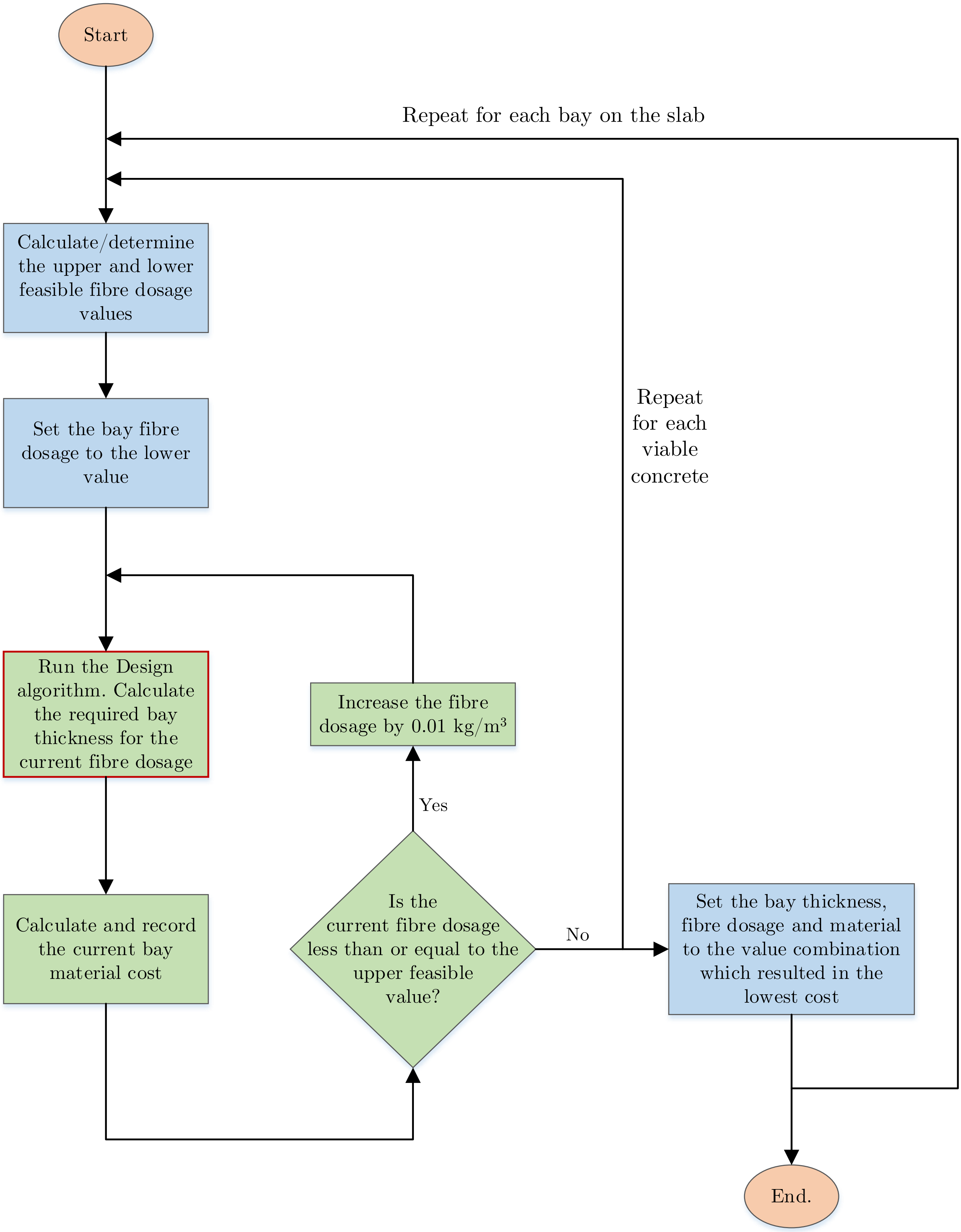


Appendix E

Flowchart for the optimised slab design procedure

The flowchart on the following page illustrates the steps performed during the optimised slab design procedure, to calculate the most cost-effective, suitable thickness and fibre-dosage combination for each bay, as outlined in Section 3.5.2.

Appendix E: Optimised slab design procedure - flowchart



Appendix F

Bay neutral axis and moment capacity formulae

A rigorous method of determining the neutral axis depth and moment capacity of a FRC bay, as given by TR34 (Concrete Society 2013), is discussed in Section 3.6.1. This method employs equations for horizontal and moment equilibrium (Equations 20 and 21, respectively). Formulae for the variables involved in these two equations are listed below (also see Equations 5 and 6).

$$\varepsilon_{ft} = \frac{3.5}{h_c} \leq 0.025 \qquad \text{Eq. F1}$$

$$h_{ux} = h - h_c \qquad \text{Eq. F2}$$

$$\varepsilon_{fc} = \frac{\varepsilon_{ft} \cdot h_{ux}}{h_c} > 0.00175 \; but \; \leq 0.0035 \qquad \text{Eq. F3}$$

$$f_{cd} = \frac{0.85 \cdot f_{ctm}}{\gamma_m} \qquad \text{Eq. F4}$$

$$\sigma_{r5} = \sigma_{r1} - \frac{\varepsilon_{ft}}{0.025}(\sigma_{r1} - \sigma_{r4}) \qquad \text{Eq. F5}$$

$$d_1 = \frac{0.00175 \cdot h_{ux}}{\varepsilon_{fc}} \qquad \text{Eq. F6}$$

$$d_2 = h_{ux} - d_1 \qquad \text{Eq. F7}$$

$$N_1 = 0.5 d_1 \cdot b \cdot f_{cd} \qquad \text{Eq. F8}$$

$$N_2 = d_2 \cdot b \cdot f_{cd} \qquad \text{Eq. F9}$$

$$T_1 = \frac{b \cdot h_c \cdot \sigma_{r5}}{\gamma_m} \qquad \text{Eq. F10}$$

$$T_2 = \frac{0.5 \cdot b \cdot h_c \cdot (\sigma_{r1} - \sigma_{r5})}{\gamma_m} \qquad \text{Eq. F11}$$

$$T_3 = \frac{A_s E_s (d - h_{ux}) \varepsilon_{fc}}{h_{ux} \cdot \gamma_s} \qquad \text{Eq. F12}$$

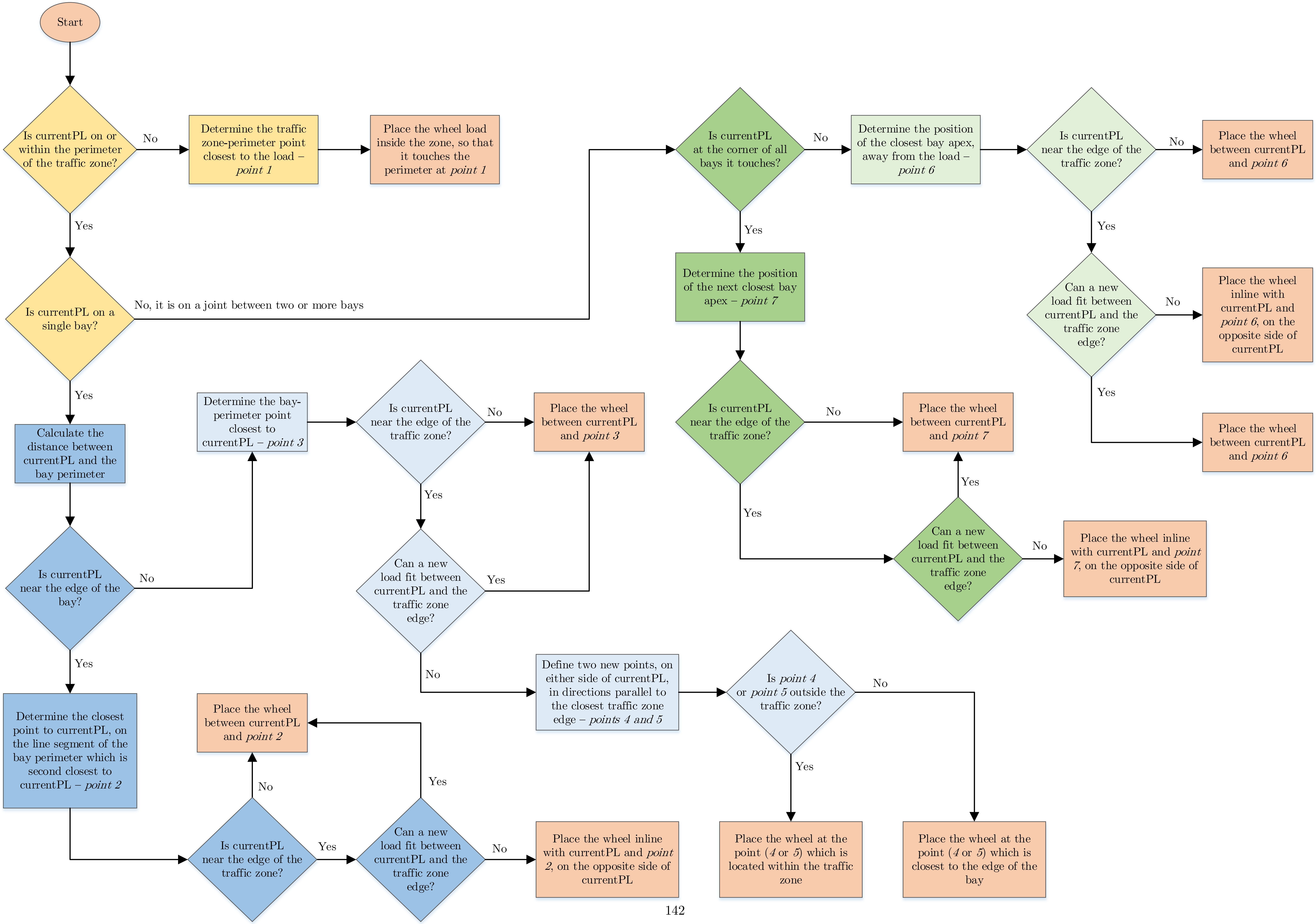
Appendix G

Placement of automatically generated traffic-zone wheel-point-loads - flowchart

The flowchart on the following page illustrates the process by which an automatically generated wheel point load is placed on a slab (see Section 4.6).

More specifically, the flowchart illustrates the placement of the first wheel load for a specific traffic zone, when considering a specific single-point-load – referred to as “currentPL” within the flowchart.

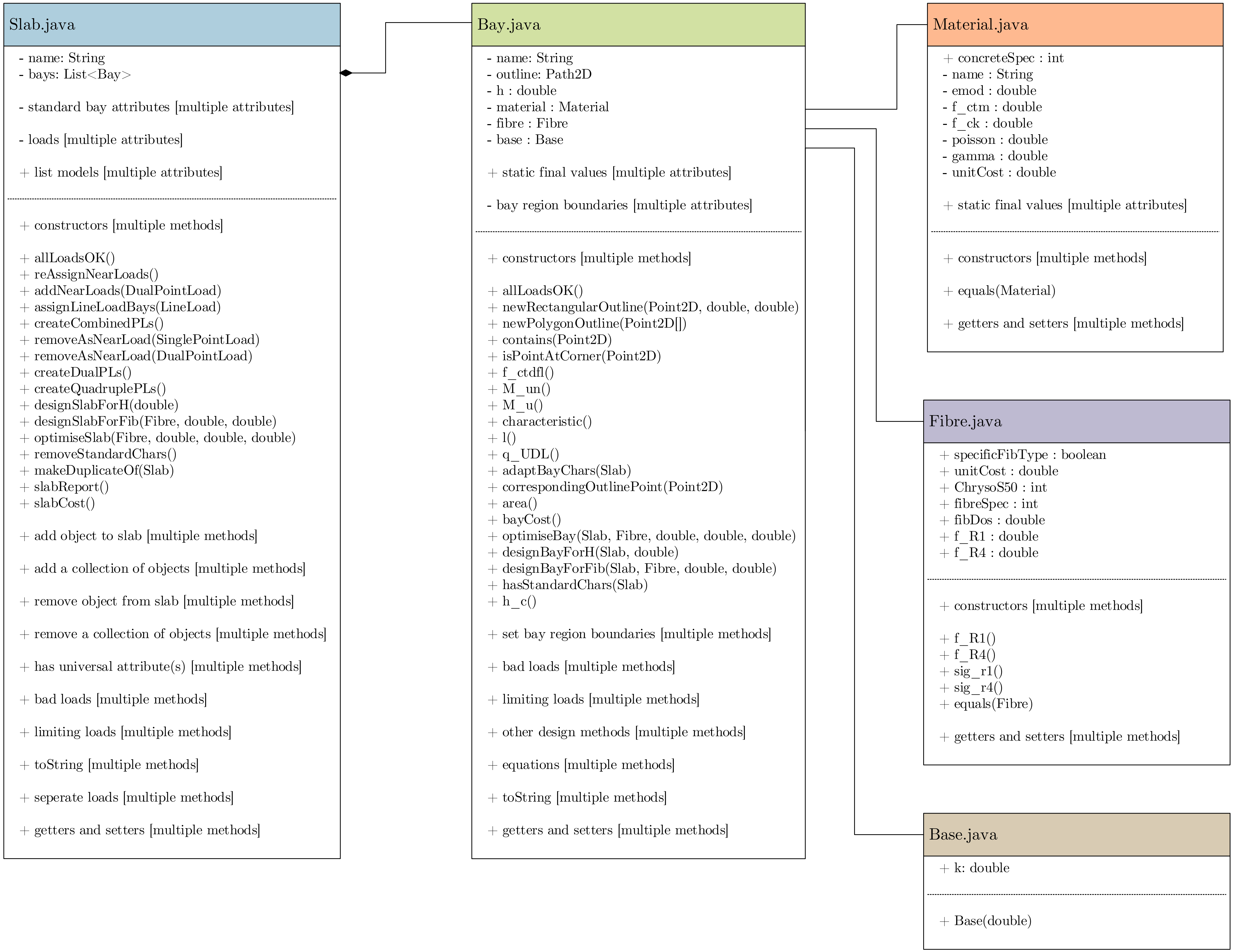
Appendix G: Placement of automatically generated traffic-zone wheel-point-loads - flowchart



Appendix H

Physical object layout for the SOG model – UML diagram

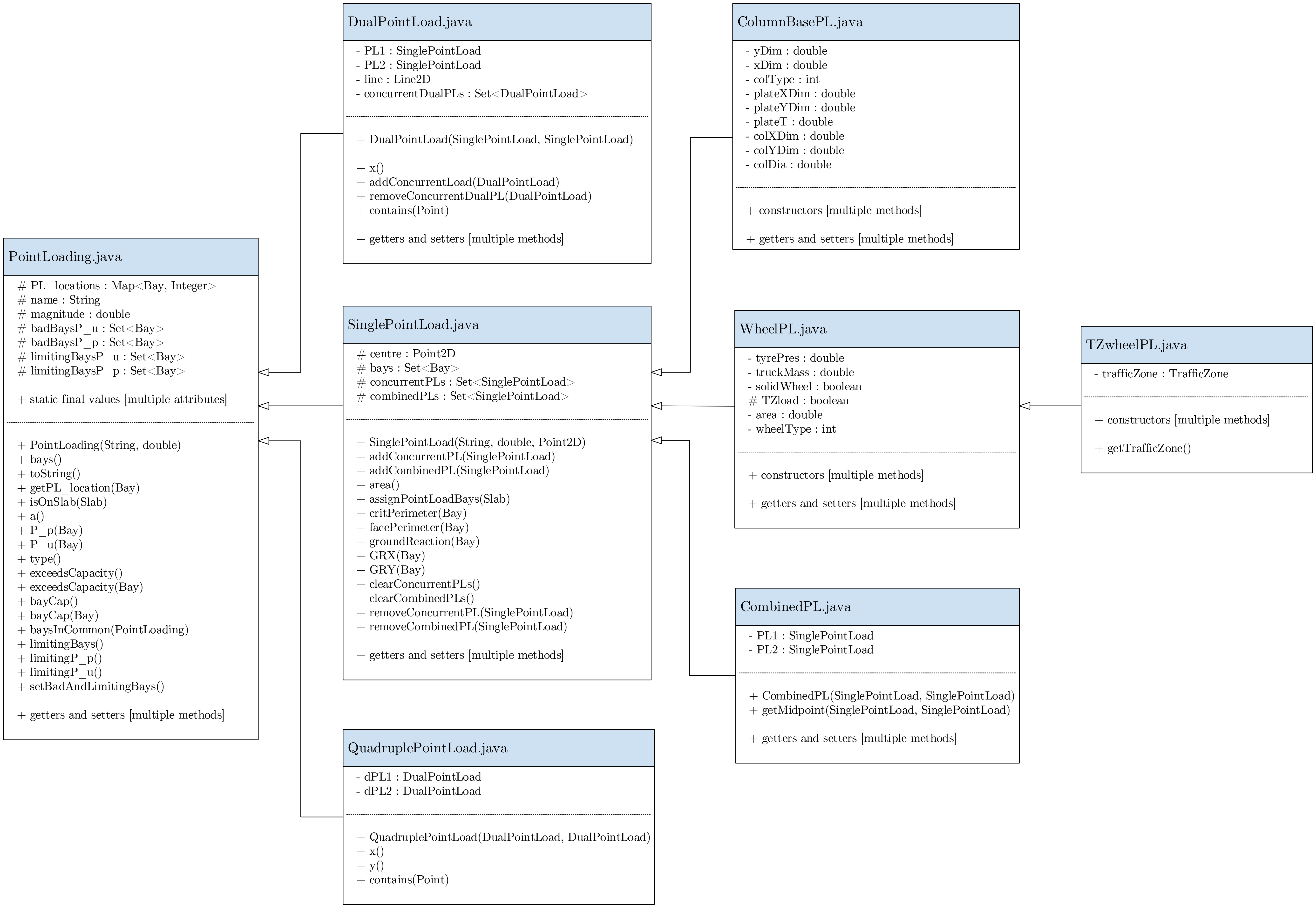
The UML diagram on the next page illustrates the relationship between the physical object classes of the fibre-reinforced SOG model, along with a summarised overview of their attributes and methods (see Section 5.1.1).


Slab.java
- name: String
- bays: List<Bay>
- standard bay attributes [multiple attributes]
- loads [multiple attributes]
+ list models [multiple attributes]
+ constructors [multiple methods]
+ allLoadsOK()
+ reAssignNearLoads()
+ addNearLoads(DualPointLoad)
+ assignLineLoadBays(LineLoad)
+ createCombinedPLs()
+ removeAsNearLoad(SinglePointLoad)
+ removeAsNearLoad(DualPointLoad)
+ createDualPLs()
+ createQuadruplePLs()
+ designSlabForH(double)
+ designSlabForFib(Fibre, double, double)
+ optimiseSlab(Fibre, double, double, double)
+ removeStandardChars()
+ makeDuplicateOf(Slab)
+ slabReport()
+ slabCost()
+ add object to slab [multiple methods]
+ add a collection of objects [multiple methods]
+ remove object from slab [multiple methods]
+ remove a collection of objects [multiple methods]
+ has universal attribute(s) [multiple methods]
+ bad loads [multiple methods]
+ limiting loads [multiple methods]
+ toString [multiple methods]
+ seperate loads [multiple methods]
+ getters and setters [multiple methods]
Bay.java
- name: String
- outline: Path2D
- h : double
- material : Material
- fibre : Fibre
- base : Base
+ static final values [multiple attributes]
- bay region boundaries [multiple attributes]
+ constructors [multiple methods]
+ allLoadsOK()
+ newRectangularOutline(Point2D, double, double)
+ newPolygonOutline(Point2D[])
+ contains(Point2D)
+ isPointAtCorner(Point2D)
+ f_ctdfl()
+ M_un()
+ M_u()
+ characteristic()
+ l()
+ q_UDL()
+ adaptBayChars(Slab)
+ correspondingOutlinePoint(Point2D)
+ area()
+ bayCost()
+ optimiseBay(Slab, Fibre, double, double, double)
+ designBayForH(Slab, double)
+ designBayForFib(Slab, Fibre, double, double)
+ hasStandardChars(Slab)
+ h_c()
+ set bay region boundaries [multiple methods]
+ bad loads [multiple methods]
+ limiting loads [multiple methods]
+ other design methods [multiple methods]
+ equations [multiple methods]
+ toString [multiple methods]
+ getters and setters [multiple methods]
Material.java
+ concreteSpec : int
- name : String
- emod : double
- f_ctm : double
- f_ck : double
- poisson : double
- gamma : double
- unitCost : double
+ static final values [multiple attributes]
+ constructors [multiple methods]
+ equals(Material)
+ getters and setters [multiple methods]
Fibre.java
+ specificFibType : boolean
+ unitCost : double
+ ChrysoS50 : int
+ fibreSpec : int
+ fibDos : double
+ f_R1 : double
+ f_R4 : double
+ constructors [multiple methods]
+ f_R1()
+ f_R4()
+ sig_r1()
+ sig_r4()
+ equals(Fibre)
+ getters and setters [multiple methods]
Base.java
+ k: double
+ Base(double)

Appendix I

Load object layout for the SOG model – UML diagram

The UML diagrams on the following two pages illustrate the relationship between the load object classes of the fibre-reinforced SOG model, along with a summarised overview of their attributes and methods (see Section 5.1.2).


DualPointLoad.java
- PL1 : SinglePointLoad
- PL2 : SinglePointLoad
- line : Line2D
- concurrentDualPLs : Set<DualPointLoad>
+ DualPointLoad(SinglePointLoad, SinglePointLoad)
+ x()
+ addConcurrentLoad(DualPointLoad)
+ removeConcurrentDualPL(DualPointLoad)
+ contains(Point)
+ getters and setters [multiple methods]
ColumnBasePL.java
- yDim : double
- xDim : double
- colType : int
- plateXDim : double
- plateYDim : double
- plateT : double
- colXDim : double
- colYDim : double
- colDia : double
+ constructors [multiple methods]
+ getters and setters [multiple methods]
PointLoading.java
PL_locations : Map<Bay, Integer>
name : String
magnitude : double
badBaysP_u : Set<Bay>
badBaysP_p : Set<Bay>
limitingBaysP_u : Set<Bay>
limitingBaysP_p : Set<Bay>
+ static final values [multiple attributes]
+ PointLoading(String, double)
+ bays()
+ toString()
+ getPL_location(Bay)
+ isOnSlab(Slab)
+ a()
+ P_p(Bay)
+ P_u(Bay)
+ type()
+ exceedsCapacity()
+ exceedsCapacity(Bay)
+ bayCap()
+ bayCap(Bay)
+ baysInCommon(PointLoading)
+ limitingBays()
+ limitingP_p()
+ limitingP_u()
+ setBadAndLimitingBays()
+ getters and setters [multiple methods]
SinglePointLoad.java
centre : Point2D
bays : Set<Bay>
concurrentPLs : Set<SinglePointLoad>
combinedPLs : Set<SinglePointLoad>
+ SinglePointLoad(String, double, Point2D)
+ addConcurrentPL(SinglePointLoad)
+ addCombinedPL(SinglePointLoad)
+ area()
+ assignPointLoadBays(Slab)
+ critPerimeter(Bay)
+ facePerimeter(Bay)
+ groundReaction(Bay)
+ GRX(Bay)
+ GRY(Bay)
+ clearConcurrentPLs()
+ clearCombinedPLs()
+ removeConcurrentPL(SinglePointLoad)
+ removeCombinedPL(SinglePointLoad)
+ getters and setters [multiple methods]
WheelPL.java
- tyrePres : double
- truckMass : double
- solidWheel : boolean
TZload : boolean
- area : double
- wheelType : int
+ constructors [multiple methods]
+ getters and setters [multiple methods]
TZwheelPL.java
- trafficZone : TrafficZone
+ constructors [multiple methods]
+ getTrafficZone()
CombinedPL.java
- PL1 : SinglePointLoad
- PL2 : SinglePointLoad
+ CombinedPL(SinglePointLoad, SinglePointLoad)
+ getMidpoint(SinglePointLoad, SinglePointLoad)
+ getters and setters [multiple methods]
QuadruplePointLoad.java
- dPL1 : DualPointLoad
- dPL2 : DualPointLoad
+ QuadruplePointLoad(DualPointLoad, DualPointLoad)
+ x()
+ y()
+ contains(Point)

LineLoad.java

- name : String
- line : Line2D
- startMag : double
- endMag : double
- baysCrossed : Set<Bay>
- critPoints : Set<Point2D>
- segments : List<LineLoadSegment>

+ LineLoad(String, Point2D, Point2D, double, double)
+ addBayCrossed(Bay)
+ assignCritPoints()
+ segmentLine()
+ magAt(Point2D)
+ onSlabString(Slab)
+ isOnSlab(Slab)
+ getMidPoint()
+ lineSegPoints()

+ getters and setters [multiple methods]

LineLoadSegment.java

- name : String
- line : Line2D
- startMag : double
- endMag : double
- bay : Bay

+ static final values [multiple attributes]

+ LineLoadSegment(String, Point2D, Point2D, double, double)
+ midPoint()
+ bayCap()
+ isOnSlab(Slab)
+ maxMag()
+ assignZone()
+ exceedsCapacity()

+ getters and setters [multiple methods]

TrafficZone.java

- name : String
- tyrePres : double
- truckMass : double
+ outPoints : Point2D[]
- perim : Path2D
- bays : Set<Bay>
- combineLoads : Set<SinglePointLoad>
- nearLoads : Set<SinglePointLoad>
- possiblePLs : Set<TZwheelPL>
+ refX : double
+ refY : double
+ xDim : double
+ yDim : double
- solidWheel : boolean
- blank : boolean
- area : double
- fillShape : boolean
- frontAxleWidth : double
- rearAxleWidth : double
- axleLength : double
- conservative : boolean

+ static final values [multiple attributes]

+ constructors [multiple methods]

- newRectangularTZ(Point2D, double, double)
- newPolygonTZ(Point2D[])
- area()
- a()
- assignTZBays(Slab)
- assignLoads(Slab)
- createComboPLs(Slab)
+ adjustWheels(TZwheelPL, TZwheelPL, SinglePointLoad)
+ worstComboPL1(SinglePointLoad)
+ worstComboPL2(SinglePointLoad, double)
+ worstComboPL3_4(TZwheelPL, TZwheelPL, SinglePointLoad)
- onSlabString(Slab)
- isOnSlab(Slab)

+ getters and setters [multiple methods]

UDL.java

+ shape : int
- name : String
- magnitude : double
- perim : Path2D
- bays : Set<Bay>
+ refX : double
+ refY : double
+ xDim : double
+ yDim : double
+ outPoints : Point2D[]
- fillShape : boolean

+ static final values [multiple attributes]

+ UDL()
+ UDL(String, double)
+ newRectangularUDL(Point2D, double, double)
+ assignUDLBays(Slab)
+ newPolygonUDL(Point2D[])
+ addBay(Bay)
+ onSlabString(Slab)
+ isOnSlab(Slab)
+ bayCap()
+ exceedsCapacity()
+ exceedsCapacity(Bay)

+ getters and setters [multiple methods]

Appendix J

Utility class layout for the SOG model – UML diagram

The UML diagram below illustrates a summarised overview of the methods of the utility class of the fibre-reinforced SOG model (see Section 5.1.3).

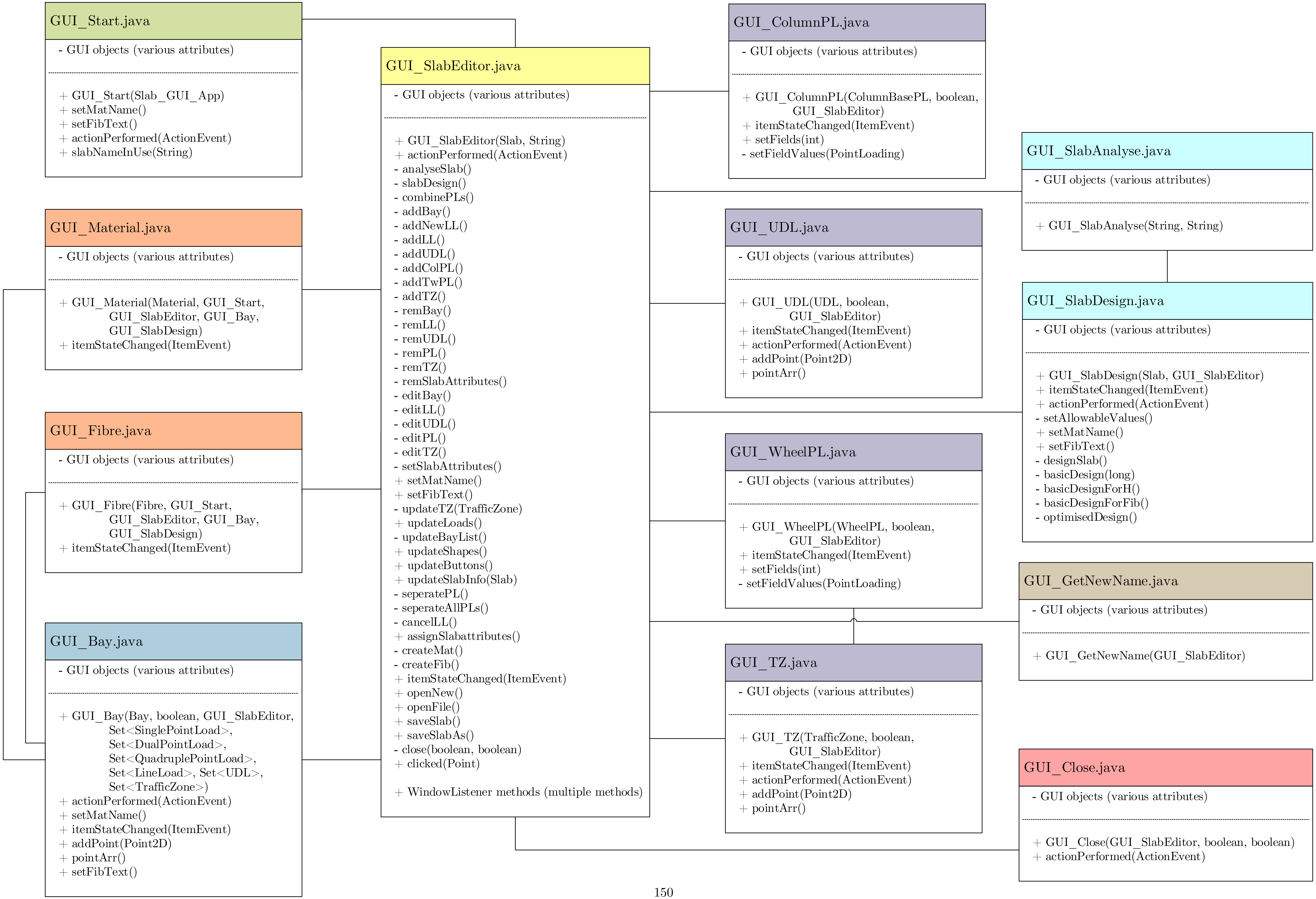
SlabUtils.java

\+ arePathsSimilar(Path2D, Path2D)
\+ bubbleSortLines(Point2D, Collection<Line2D>)
\+ closestLineSegs(Path2D, Point2D)
\+ closestPerimApex(Path2D, Point2D)
\+ closestPerimPoint(Path2D, Point2D)
\+ combinedLine(Line2D, Line2D)
\+ connectedLines(Point2D, Path2D)
\+ contains(Line2D, Point2D)
\+ contains(Path2D, Point2D)
\+ contains(Path2D, Ellipse2D)
\+ contains(Path2D, Path2D)
\+ distance(Line2D, Point2D)
\+ distance(Path2D, Line2D)
\+ distance(Path2D, Point2D)
\+ getInputFile()
\+ getOutputFile()
\+ internalParallelPath(Path2D, double)
\+ isPathDefined(Path2D)
\+ lineExtendsLine(Line2D, Line2D)
\+ linesConnectedAtEnds(Line2D, Line2D)
\+ listToString(List<Line2D>)
\+ path(Point2D[])
\+ pathPoints(Path2D)
\+ pathLines(Path2D)
\+ pathToString(Path2D)
\+ pathSelfCollapses(Path2D)
\+ pathSelfIntersects(Path2D)
\+ pathsIntersect(Path2D, Path2D)
\+ projection(Point2D, Line2D)
\+ roundUpToInt(double)
\+ secondClosestLineSeg(Path2D, Point2D)
\+ secondClosestPerimPoint(Path2D, Point2D)
\+ secondClosestPerimApex(Path2D, Point2D)

Appendix K

GUI object layout for the SOG model – UML diagram

The UML diagram on the following page illustrates the relationship between the GUI object classes of the fibre-reinforced SOG model, along with a summarised overview of their attributes and methods (see Section 5.1.4).


GUI_Start.java
- GUI objects (various attributes)
+ GUI_Start(Slab_GUI_App)
+ setMatName()
+ setFibText()
+ actionPerformed(ActionEvent)
+ slabNameInUse(String)
GUI_Material.java
- GUI objects (various attributes)
+ GUI_Material(Material, GUI_Start, GUI_SlabEditor, GUI_Bay, GUI_SlabDesign)
+ itemStateChanged(ItemEvent)
GUI_Fibre.java
- GUI objects (various attributes)
+ GUI_Fibre(Fibre, GUI_Start, GUI_SlabEditor, GUI_Bay, GUI_SlabDesign)
+ itemStateChanged(ItemEvent)
GUI_Bay.java
- GUI objects (various attributes)
+ GUI_Bay(Bay, boolean, GUI_SlabEditor, Set<SinglePointLoad>, Set<DualPointLoad>, Set<QuadruplePointLoad>, Set<LineLoad>, Set<UDL>, Set<TrafficZone>)
+ actionPerformed(ActionEvent)
+ setMatName()
+ itemStateChanged(ItemEvent)
+ addPoint(Point2D)
+ pointArr()
+ setFibText()
GUI_SlabEditor.java
- GUI objects (various attributes)
+ GUI_SlabEditor(Slab, String)
+ actionPerformed(ActionEvent)
- analyseSlab()
- slabDesign()
- combinePLs()
- addBay()
- addNewLL()
- addLL()
- addUDL()
- addColPL()
- addTwPL()
- addTZ()
- remBay()
- remLL()
- remUDL()
- remPL()
- remTZ()
- remSlabAttributes()
- editBay()
- editLL()
- editUDL()
- editPL()
- editTZ()
- setSlabAttributes()
+ setMatName()
+ setFibText()
- updateTZ(TrafficZone)
+ updateLoads()
- updateBayList()
+ updateShapes()
+ updateButtons()
+ updateSlabInfo(Slab)
- seperatePL()
- seperateAllPLs()
- cancelLL()
+ assignSlabattributes()
- createMat()
- createFib()
+ itemStateChanged(ItemEvent)
+ openNew()
+ openFile()
+ saveSlab()
+ saveSlabAs()
- close(boolean, boolean)
+ clicked(Point)
+ WindowListener methods (multiple methods)
GUI_ColumnPL.java
- GUI objects (various attributes)
+ GUI_ColumnPL(ColumnBasePL, boolean, GUI_SlabEditor)
+ itemStateChanged(ItemEvent)
+ setFields(int)
- setFieldValues(PointLoading)
GUI_UDL.java
- GUI objects (various attributes)
+ GUI_UDL(UDL, boolean, GUI_SlabEditor)
+ itemStateChanged(ItemEvent)
+ actionPerformed(ActionEvent)
+ addPoint(Point2D)
+ pointArr()
GUI_WheelPL.java
- GUI objects (various attributes)
+ GUI_WheelPL(WheelPL, boolean, GUI_SlabEditor)
+ itemStateChanged(ItemEvent)
+ setFields(int)
- setFieldValues(PointLoading)
GUI_TZ.java
- GUI objects (various attributes)
+ GUI_TZ(TrafficZone, boolean, GUI_SlabEditor)
+ itemStateChanged(ItemEvent)
+ actionPerformed(ActionEvent)
+ addPoint(Point2D)
+ pointArr()
GUI_SlabAnalyse.java
- GUI objects (various attributes)
+ GUI_SlabAnalyse(String, String)
GUI_SlabDesign.java
- GUI objects (various attributes)
+ GUI_SlabDesign(Slab, GUI_SlabEditor)
+ itemStateChanged(ItemEvent)
+ actionPerformed(ActionEvent)
- setAllowableValues()
+ setMatName()
+ setFibText()
- designSlab()
- basicDesign(long)
- basicDesignForH()
- basicDesignForFib()
- optimisedDesign()
GUI_GetNewName.java
- GUI objects (various attributes)
+ GUI_GetNewName(GUI_SlabEditor)
GUI_Close.java
- GUI objects (various attributes)
+ GUI_Close(GUI_SlabEditor, boolean, boolean)
+ actionPerformed(ActionEvent)

Appendix L

Unit test reports for GUI components

As discussed in Section 7.1, tables L.1 to L.13 outline the series of unit tests which have been completed to verify the correct execution of all GUI functionalities.

Table L.1: Unit test report- Welcoming window

Functionality	**Checked**
Open an existing ".slb" file	✓
Display the file selector window	✓
Display an error message if no file has been selected	✓
Display an error message if the selected file not of the correct format	✓
Create slab without standard attributes	✓
Set certain (not all) standard slab attributes	✓
Set all standard slab attributes	✓
Create slab with standard attributes	✓
Display error message if no slab name is entered	✓
Display error message if slab attributes are entered in the wrong format	✓

Table L.2: Unit test report- Slab editor window

Tab/ Panel	**Functionality**	**Checked**
General	Display a drawing of the slab, with all bays and loads. Updated in real time.	✓
	Perform a slab analysis of the existing slab situation (open the Slab Analysis window)	✓
	Perform a slab design for the existing slab situation (open the Slab Design window)	✓
	If the user clicks on the drawing, proceed to the selected load or slab object	✓
	Display an error message if the user attempts to perform a slab analysis before any bays have been created.	✓
	Display an error message if the user attempts to perform a slab design before any bays have been created.	✓

Slab tab	If any standard slab attributes have been assigned, display all relevant field values	✓
	Assign standard slab attributes (if none have been assigned)	✓
	Re-assign/edit existing standard slab attributes	✓
	Remove existing standard slab attributes	✓
	Display a list of bays which currently make up the slab	✓
	Add a new bay to the slab (open the Bay editor window)	✓
	Remove a selected bay from the slab	✓
	When a bay is removed, also remove loads which fall only on the bay	✓
	Edit a selected bay (open the Bay editor window with current bay data)	✓
	Display/don't display all the respective bay lines, depending on the user's selection, in real time.	✓
	Display/don't display the slab in colour, depending on the user's selection, in real time.	✓
Point load tab	Display a list of point loads currently on the slab	✓
	Add a new point load to the slab (open the point load editor window)	✓
	Remove a selected point load from the slab	✓
	Edit a selected point load (open the Point load editor window with current point load data)	✓
	Combine point loads to form new Dual Point loads and Quadruple Point loads, as applicable	✓
	Separate the selected combined load	✓
	Separate all combined loads	✓
	Display an error message if the user attempts to separate a single point load	✓
	Display an error message if the user attempts to add a point load before any bay(s) have been created	✓
	Inform the user if a point load is not added to the slab because it is not located on the slab	✓
Line load tab	Display a list of line loads currently on the slab	✓
	Add a new line load to the slab (display the line load editor panel)	✓
	Remove a selected line load from the slab	✓
	Edit a selected line load (open the line load editor panel with current line load data)	✓
	Display an error message if the user attempts to add a line load before any bay(s) have been created	✓
	Display an error message if any of the input line load values are not entered or are in the wrong format	✓
	Display an error message if Point 1 and Point 2 of the line have identical coordinates	✓
	Show or hide line load segments on the slab drawing	✓

	Cancel line load input (do nothing and hide the panel)	✓
	Create a line load which is entirely on a single bay	✓
	Create a line load which runs over multiple bays	✓
	Create a line load which runs off of the slab	✓
	Display a warning message if the user adds a line load which is not entirely on the slab	✓
	Inform the user if a line load is not added to the slab because it is not located on the slab	✓
	Create a line load which starts/stops on a bay edge/joint	✓
	Create a line load which starts/stops on a bay corner	✓
	Create a line load which runs on a bay edge/joint	✓
	If the line load already exists, display all relevant field values	✓
Uniform distributed load tab	Display a list of UDLs currently on the slab	✓
	Add a new UDL to the slab (display the UDL editor window)	✓
	Remove a selected UDL from the slab	✓
	Edit a selected UDL (open the UDL editor window with current UDL data)	✓
	Display a warning message if the user adds a UDL which runs off the edge of the slab	✓
	Inform the user if a UDL is not added to the slab because it is not located on the slab	✓
Traffic zones tab	Display a list of traffic zones currently on the slab	✓
	Add a new traffic zone to the slab (open the Traffic zone editor window)	✓
	Remove a selected traffic zone from the slab	✓
	Edit a selected traffic zone (open the Traffic zone editor window with current traffic zone data)	✓
	Display a warning message if the user adds a traffic zone which runs off the edge of the slab	✓
	Inform the user if a traffic zone is not added to the slab because it is not located on the slab	✓
	Update (add or remove) traffic zone loads, based on other loads on the slab	✓

Table L.3: Unit test report- Material editor window

Functionality	**Checked**
Create a material from a list of pre-existing concrete types	✓
Create a material by specifying all the relevant material properties	✓
Only allow the user to edit information which is relevant to the type of material selected	✓
Display an error message if "Specify concrete properties" is selected, but all values are not entered or if any value is in the wrong format	✓
Cancel material input (do nothing and exit window)	✓
If the material already exists, display all relevant field values	✓

Table L.4: Unit test report- Fibre editor window

Functionality	**Checked**
Create a fibre object by specifying the a fibre type (specific product) and dosage	✓
Create a fibre object by specifying the f_{R1} and f_{R4} values of the material	✓
Only allow the user to edit information which is relevant to the type of fibre input selected	✓
Display an error message if "Specific fibre product" is selected and fibre dosage is in the wrong format	✓
Display an error message if "Unknown fibre product" is selected and values for f_{R1} and/or f_{R4} are in the wrong format	✓
Cancel fibre input (do nothing and exit window)	✓
If the fibre already exists, display all relevant field values	✓

Table L.5: Unit test report- Bay editor window

Functionality	**Checked**
Assign bay attributes (if none have been assumed from standard slab attributes)	✓
Re-assign/edit existing bay attributes	✓
Display an error message if the user attempts to create/update a bay without defining it's outline	✓
Only allow the user to edit information which is relevant to the selected bay shape	✓
Display an error message if "Rectangular bay" is selected, but all values are not entered or if any value is in the wrong format	✓
Display an error message if "Polygonal bay" is selected, but values for X and Y are not entered or if either value is in the wrong format	✓
Display an error message if "Polygonal bay" is selected, but less than three points have been entered	✓
Display a list of points which has been added to the polygonal bay perimeter	✓
Add a new point (with X and Y coordinates) to the polygonal bay perimeter	✓
Remove a selected point from the polygonal bay perimeter	✓
Cancel bay input (do nothing and exit window)	✓
If the bay already exists, display all relevant field values	✓

Table L.6: Unit test report- Column base point load editor window

Functionality	Checked
Display an error message if all necessary values are not entered or if any value is in the wrong format	✓
Create a point load on the joint between two bays	✓
Create multiple point loads if the "Add multiple identical column loads" check box is selected and all input values are acceptable	✓
Inform the user if a point load is not added to the slab because it is not located on the slab	✓
Cancel point load input (do nothing and exit window)	✓
Assign conservative values if plate and/or column dimensions are not entered	✓
Only allow the user to edit information which is relevant to the selected column shape	✓
Display an error message if "Square column" is selected and the applicable measurements are not entered or are entered in the wrong format	✓
Display an error message if the width, length or diameter of a column exceeds the plate width or length	✓
Display an error message if "Round column" is selected and the applicable measurements are not entered or are entered in the wrong format	✓

Table L.7: Unit test report- Truck wheel point load editor window

Functionality	Checked
Display an error message if all necessary values are not entered or if any value is in the wrong format	✓
Create a point load on the joint between two bays	✓
Inform the user if a point load is not added to the slab because it is not located on the slab	✓
Cancel point load input (do nothing and exit window)	✓
Display an error message if "Pneumatic tyres" is selected and Tyre pressure is not entered or is in the wrong format	✓
Display an error message if "Solid tyres" is selected and Contact area is not entered or is in the wrong format	✓
Display an error message if no tyre type is selected	✓

Table L.8: Unit test report- UDL editor

Functionality	Checked
Display an error message if the user attempts to create/update a UDL without defining it's outline	✓
Only allow the user to edit information which is relevant to the selected UDL shape	✓
Display an error message if "Rectangular UDL" is selected, but all values are not entered or if any value is in the wrong format	✓

Display an error message if "Polygonal UDL" is selected, but values for X and Y are not entered or if either value is in the wrong format	✓
Display an error message if "Polygonal UDL" is selected, but less than three points have been entered	✓
Display a list of points which has been added to the polygonal UDL perimeter	✓
Add a new point (with X and Y coordinates) to the polygonal UDL perimeter	✓
Remove a selected point from the polygonal UDL perimeter	✓
Cancel UDL input (do nothing and exit window)	✓
If the UDL already exists, display all relevant field values	✓

Table L.9: Unit test report- Traffic zone editor window

Functionality	**Checked**
Display an error message if the user attempts to add a traffic zone before any bay(s) have been created	✓
Display an error message if the user attempts to create/update a Traffic Zone without defining it's outline	✓
Only allow the user to edit information which is relevant to the selected Traffic Zone shape	✓
Display an error message if "Rectangular traffic zone" is selected, but all values are not entered or if any value is in the wrong format	✓
Display an error message if "Polygonal traffic zone" is selected, but values for X and Y are not entered or if either value is in the wrong format	✓
Display an error message if "Polygonal traffic zone" is selected, but less than three points have been entered	✓
Display a list of points which has been added to the polygonal Traffic Zone perimeter	✓
Add a new point (with X and Y coordinates) to the polygonal Traffic Zone perimeter	✓
Remove a selected point from the polygonal Traffic Zone perimeter	✓
Cancel Traffic Zone input (do nothing and exit window)	✓
If the Traffic Zone already exists, display all relevant field values	✓
Display an error message if "Pneumatic tyres" is selected and Tyre pressure is not entered or is in the wrong format	✓
Display an error message if "Solid tyres" is selected and Contact area is not entered or is in the wrong format	✓
Display an error message if no tyre type is selected	✓

Table L.10: Unit test report- Slab analysis window

Functionality	**Checked**
Compile and display a slab report for the current slab situation	✓
Create a PDF document of the slab report	✓
Cancel slab analysis (do nothing and exit window)	✓

Table L.11: Unit test report- Slab design window

Functionality	**Checked**
Perform an optimised slab design	✓
Perform a slab design by varying slab/bay thickness(es), maintaining current fibre content(s).	✓
Perform a slab design by varying slab/bay fibre content(s), maintaining current thickness(es).	✓
Display the correct and applicable information in all text fields	✓
Standardise selected attributes across all bays	✓
After performing the selected design, perform a slab analysis (open the Slab Analysis window)	✓
Display an error message if any required value is omitted or unacceptable	✓
Cancel slab design (do nothing and exit window)	✓

Table L.12: Unit test report- Slab rename window

Functionality	**Checked**
Rename the slab	✓
Cancel slab rename (do nothing and exit window)	✓

Table L.13: Unit test report- Program exit window

Functionality	**Checked**
Save the slab object and exit the program	✓
Exit the program without saving	✓
Cancel program exit (do nothing and exit window)	✓